HIGH SPEED DATA ACQUISITION SYSTEM

Mack Taylor Elliott

NAVAL POSTGRADUATE SCHOOL
Monterey, California

THESIS

HIGH SPEED DATA ACQUISITION SYSTEM

by

Mack Taylor Elliott

September 1978

Thesis Advisor: Louis V. Schmidt

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

REPORT DOCUMENTATION PAGE		READ INSTRUCTIONS BEFORE COMPLETING FORM
1. REPORT NUMBER	2. GOVT ACCESSION NO.	3. RECIPIENT'S CATALOG NUMBER
4. TITLE (and Subtitle) HIGH SPEED DATA ACQUISITION SYSTEM		5. TYPE OF REPORT & PERIOD COVERED Master's Thesis; Sept 1978
		6. PERFORMING ORG. REPORT NUMBER
7. AUTHOR(s) Mack Taylor Elliott		8. CONTRACT OR GRANT NUMBER(s)
9. PERFORMING ORGANIZATION NAME AND ADDRESS Naval Postgraduate School Monterey, California 93940		10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS
11. CONTROLLING OFFICE NAME AND ADDRESS Naval Postgraduate School Monterey, California 93940		12. REPORT DATE September 1978
		13. NUMBER OF PAGES 191
14. MONITORING AGENCY NAME & ADDRESS(if different from Controlling Office) Naval Postgraduate School Monterey, California 93940		15. SECURITY CLASS. (of this report) Unclassified
		15a. DECLASSIFICATION/DOWNGRADING SCHEDULE

16. DISTRIBUTION STATEMENT (of this Report)

Approved for public release; distribution unlimited

17. DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report)

18. SUPPLEMENTARY NOTES

19. KEY WORDS (Continue on reverse side if necessary and identify by block number)

analog to digital converter
data acquisition
microcomputer
microprocessor
unsteady aerodynamics

20. ABSTRACT (Continue on reverse side if necessary and identify by block number)

This paper describes the expansion and modification of an existing data acquisition system to effect extensive improvements in speed and flexibility. A microprocessor, flexible disk drive, analog to digital converter, direct memory access module, and high-speed line printer were integrated and interfaced to an IBM 360 digital computer with a high-speed data transmission line.

The resultant system provided the capability of digitizing up to

DD FORM 1 JAN 73 1473 EDITION OF 1 NOV 65 IS OBSOLETE
S/N 0102-014-6601 |

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

sixteen analog inputs simultaneously at rates in excess of 45,000 samples per second. The experimental data could be transmitted expeditiously to the IBM 360 computer for efficient manipulation. Additional benefits gained from the system were its capabilities as a remote terminal for the IBM 360 and a typewriter-quality word processor. The data acquisition and reduction system was qualified for functional performance and speed through a series of test exercises. The word processor was demonstrated in the production of this document.

Approved for public release; distribution unlimited

HIGH SPEED DATA ACQUISITION SYSTEM

by

Mack Taylor Elliott
Lieutenant, United States Navy
B. S., University of South Carolina 1971

Submitted in partial fulfillment of the
requirements for the degree of

MASTER OF SCIENCE IN AERONAUTICAL ENGINEERING

from the

NAVAL POSTGRADUATE SCHOOL

September 1978

ABSTRACT

This paper describes the expansion and modification of an existing data acquisition system to effect extensive improvements in speed and flexibility. A microprocessor, flexible disk drive, analog to digital converter, direct memory access module, and high-speed line printer were integrated and interfaced to an IBM 360 digital computer with a high-speed data transmission line.

The resultant system provided the capability of digitizing up to sixteen analog inputs simultaneously at rates in excess of 45,000 samples per second. The experimental data could be transmitted expeditiously to the IBM 360 computer for efficient manipulation. Additional benefits gained from the system were its capabilities as a remote terminal for the IBM 360 and a typewriter-quality word processor. The data acquisition and reduction system was qualified for functional performance and speed through a series of test exercises. The word processor was demonstrated in the production of this document.

TABLE OF CONTENTS

LIST OF FIGURES

ACKNOWLEDGEMENTS

The author wishes to express his gratitude to the many individuals who contributed their time and efforts to this project. LCDR Duane Englehardt and Ted Dunton, both of the Department of Aeronautics, and Kathryn Strutynski of the W. R. Church Computer Center furnished untiring assistance with technical problems and programming. Professor Louis V. Schmidt of the Department of Aeronautics provided guidance and expertise during all phases of development. Also, a very special thanks is due to Betty Ann, Jeremy, and Joshua for their support and infinite patience.

I. INTRODUCTION

The advent of the low-cost microprocessor system has made possible the conduct of numerically controlled laboratory experiments such as described by Casko, Ref. 11. An equally important application is in data acquisition and analysis. The ability of the microprocessor to accommodate many different tasks by software (program) changes has resulted in a very flexible system for an academic laboratory environment. Because of the relatively low cost of a complete microprocessor system, which should more aptly be termed a microcomputer, it is now practical to do experiments in aeronautics with an orientation toward investigating unsteady or time varying physical behavior. Recent experiments on the Circulation Controlled Airfoil, as reported by Englehardt in Ref. 1, are an example of the type of work which can be done economically in establishing the frequency response behavior of aerodynamic configurations.

In improving the experimental capabilities of the microcomputer system in the Department of Aeronautics, several features became evident as desired goals. Included in these goals were:

A. To extend the useful frequency range for data acquisition by verifying the Analog to Digital (A/D) sampling rate potential of an existing system data card as being on the order of 40,000 samples per second for situations of routine usage.

B. To upgrade the use of output printing devices to a typewriter-quality line printer with a maximum output baud rate of 9600 in comparison to the more common Teletype Model ASR-33 or ASR-35 baud rate of 110.

C. Although the microcomputer system had an internal computational package allowing the option of software programming for data reduction in BASIC language, it was desirable to link the microcomputer system to the IBM 360/67 digital computer at the W. R. Church Computer Center for increasing the scope (both complexity and speed) of data reduction for digitized data sets.

This thesis describes the approaches taken to achieve the above stated goals in order to improve both system flexibility and computational speed while retaining the advantages of local autonomy and cost effectiveness provided by the use of a microcomputer system.

II. HARDWARE

The original concept of the microcomputer or microprocessor involved the design of a low-cost compact version of the large digital computers. According to Osborne, Ref. 12, the resultant design differed from the goal primarily due to the distribution of logic on integrated circuit chips. Some differences in addressing modes and execution times were evident in the microcomputers.

The system used in this project had a sixteen line address bus capable of addressing 65,536 locations (2 to the 16th power). Data processed by the microcomputer travelled over an eight line data bus. The data bus is capable of handling eight binary digits (bits), or one byte, at a time. Similarly the central processor unit (CPU) within the microcomputer can work with only one byte at a time. Although sixteen bit CPU's and data busses have recently been developed, the large number of existing eight bit CPU chips assures us that the eight bit bus will be in usage for quite some time.

Subsequently data processing or numerical manipulation in the eight bit system is a relatively slow and pedestrian process. Numerical accuracy requires representing a number

by several bytes, and in much of our software the floating point binary number is represented by four bytes consisting of exponent, sign, and magnitude. Long cumbersome algorithms manipulate one byte at a time and then collocate the individual results into one total number. The addition of a peripheral device specializing in numerical manipulation, called a "math pack", can expedite the process considerably. However, all input/output operations would still be limited by the eight binary parallel digit capacity of the CPU and data bus.

The approach taken in this thesis was to avoid, to the greatest extent possible, any data manipulation by the microprocessor and instead to use it only as a control for faster peripheral devices. The data manipulation was then accomplished with the IBM 360 digital computer.

A. Components

The major components utilized in the project are discussed briefly in this thesis, and detailed descriptions are given in the referenced material. Because of the inherent complexity of integrated circuitry and digital logic considerations, even the reference manuals are often incomplete. Ignorance of a subtle but important detail about a particular component can cause the neophyte student of microprocessor technology to make errors which are

difficult to identify and cause unpredictable results. Emphasis has been put, therefore, on identifying particular idiosyncracies which have been exposed during this project and hopefully the errors need not be repeated.

1. Microprocessor

The Intel MDS-800 Microcomputer Development System with central processor unit, 64K of random access memory, front panel controller, and mainframe enclosure has been documented extensively in Ref. 1. The MDS-800 and connected flexible disk drives, CRT terminal, and paper tape reader were the benchmark devices for the project. The system, although not quite state-of-the-art in terms of microprocessors, was nevertheless a well-developed and popular system for which substantial software had been developed.

2. Analog to Digital Converter

The Datel Sinetrac-800 Analog to Digital Converter, also described in Ref. 1, was reconfigured according to the specifications in Ref. 2 for use in the Direct Memory Access (DMA) mode. Basically the only changes necessary were disabling the address structure to prevent the CPU from writing to the converter directly, and enabling the circuit board for DMA operation. Parameters left unchanged included the input voltage range of +/- five volts, twelve bit reso-

lution, twos complement output coding with sign extension, and the scan-clock option enabled. The converter digitized each analog signal into two bytes which required two memory locations. The least significant twelve bits provided a resolution of two to the 12th power (4Ø96). When applied to the input voltage range, this resolution meant an accuracy of +/- Ø.ØØ2 volts. The remaining four bits of the digitized input formed a hex digit, either Ø or F, which represented a positive or negative sign. Connection of the external analog inputs to the converter was made via a locally prepared terminal box.

Several options were available for determining the scan repetition rate. The scan-clock option allowed for a hardware variable scan rate but did not provide enough flexibility. Another possibility was to use software control through the CPU but this option was too slow. An approach which provided a greater degree of flexibility utilized the SBC Intel 534 Input/Output board to time the scan intervals, and involved operating the ST-8ØØ on an interrupt basis so the interrupt structure was enabled. The final configuration, however, excluded interrupts by the device, hence the interrupt logic wiring was again disabled.

3. Direct Memory Access

The Intel SBC-5Ø1 Direct Memory Access (DMA) Channel

Controller board was utilized to greatly decrease the throughput time of analog signal to memory storage. As reported in Ref. 1, the analog to digital converter, when operated under direct program control, had a throughput time of 76.5 microseconds per channel. This relatively slow rate was caused by the necessity of multiple transfers of each word of converted data from converter to CPU to memory with each transfer requiring several time-consuming commands to be issued by the CPU.

According to the specifications in Ref. 3, the DMA controller board was configured for base address and interrupt level and installed in the MDS-800 mainframe. A wiring harness obtained from the Datel Corporation connected the DMA board to the ST-800 converter. The DMA was programmed by the CPU to transfer a specific number of data words from the converter directly to random access memory. Then control of the data bus was relinquished by the CPU and the DMA and ST-800 were allowed to work together at maximum speed. Using full handshaking to avoid data overruns, the ST-800 sampled and converted analog signals which were routed through the DMA directly into memory. The CPU was bypassed and consequently the throughput time was reduced to 21.7 microseconds. Utilization of a pulse generator to initiate each scan gave total flexibility to the data sampling rate within the outside limit of 45,000 Hertz.

4. High-speed Printer

The Teletype Model 40 Printer was chosen to supplement the teletype terminal used in earlier projects. The Model 40 is a chain-type printer capable of 9600 baud (or 960 characters per second). Upper and lower case letters are available as is the option to use a variety of paper sizes. The printer was interfaced through a serial transmission Universal Synchronous Asynchronous Receiver Transmitter (USART) on the Intel SBC 534 board and programmed to use the standard 11 X 14 inch paper stock. Switch selectable options on the printer were set as desired in accordance with Ref. 4. The major problem that occurred when interfacing the printer was an incorrectly wired interconnector in the printer enclosure.

5. Full-sized Digital Computer

The International Business Machines Model 360/67, located in the W. R. Church Computer Center, was interfaced to the microprocessor via an RS-232C driver and telephone line. The interface, called a "high-speed line" because of its improved speed of transmission over earlier connections, was also serially driven by a USART on the SBC 534 board. Operating at baud rate of 1200 baud, the interface provided the capability of transmitting data to the larger computer which was designed for more efficient data manipulation.

The line from the microprocessor fed into the IBM 360 through an IBM 2701 Data Adapter unit controlled by the Control Program-67/Cambridge Monitoring System. Interface requirements that were imposed by the IBM 2701 were obtained from Ref. 5.

B. Interfaces

The Intel SBC 534 Four Channel Communications Expansion Board, described in Ref. 6, was used to interface the microprocessor with both the printer and the high-speed line. The SBC 534 board was selected because of the flexibility it afforded with regard to future improvements to the system. The board was jumper configured for base address, installed in the MDS-800 mainframe, and connected to the high-speed line and printer by locally prepared wiring harnesses. Two of four serial 8251 USART's and two of six programmable timer circuits on the board were utilized for the interfaces. One Programmable Interrupt Controller (PIC) of two on the board was used in an alternate approach mentioned later, but the final configuration left the PIC disabled. Another circuit available on the board for future use is an 8255 Programmable Peripheral Interface. Exact specifications and operational descriptions of the individual circuits on the SBC 534 board were found in Refs. 7 and 8.

The rates of transmission and reception of data by the

USART's were determined by the programmable timer circuits. The timers were software programmed with the appropriate countdown number and effectively divided the master clock frequency of 1.2288 Megahertz by that countdown number. The outputs of the timer circuits were jumper connected to the Transmit Clock (TxC) and Receive Clock (RxC) pins on the respective USART's.

1. Printer Interface

The Teletype Model 40 Printer interface required the consideration of handshaking signals between the USART's on the SBC 534 board and printer to maximize the speed of transmission while avoiding any data overrun. Connections between the SBC 534 and Model 40 were as indicated in Fig. 1. A command issued by the CPU to the SBC 534 USART caused the Data Transmit Ready (DTR) to go high, thus turning on the printer motor. Whenever the printer was Ready for Next Character (RNC), the Clear To Send (CTS) line on the USART enabled the Transmit Data (TxD) function. If the printer ran out of paper, the RNC line went low until the condition was rectified. Since data transmission was one-way from microprocessor to printer, other handshaking facilities were not needed.

SBC 534		Model 40
DTR	------→	DSR (motor on)
CTS	←------	RNC
TxD	------→	RxD
Grd	--------	Grd

Figure 1 - Handshaking on the Model 40 Printer

Of two one-byte data buffers involved in the transmit function of the USART, one actually transmitted the data words serially (similar in operation to a shift register). This action was enabled by the CTS line indicating that the printer was ready to receive. The second buffer accepted data words from the CPU and loaded the first buffer in parallel at the proper time. The full or empty condition of the second buffer could be determined during program control by checking the value of the Transmitter Empty (TxE) bit in the USART status word.

2. High-speed Line Interface

In the high-speed line interface, there was no handshaking between the SBC 534 USART and the IBM 2701 unit. The only hardware consideration was how to enable the Clear To Send (CTS) line on the USART. By permanently connecting the Data Transmit Ready (DTR) and CTS lines on the USART, the CTS and thus the transmit data (TxD) function were enabled by setting the DTR bit to high in the command word from the CPU to the USART. The

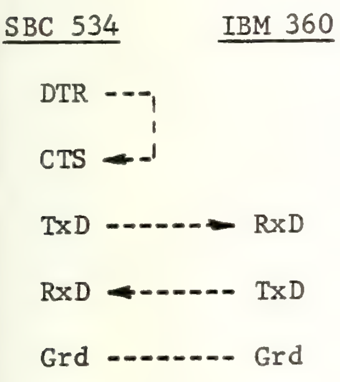


Figure 2 - Handshaking on the high-speed line

obvious problems associated with the absence of handshaking were solved through software provisions.

3. Analog to Digital Converter Interface

The ST-800 converter was already configured except for minor changes to accommodate DMA operation. It was installed in the MDS-800 mainframe, and connected to the SBC-501 DMA controller board and the analog input terminal. All handshaking between the ST-800 and DMA controller was automatic as described in Ref. 2.

The scan-clock option, which provided for a selectable delay between scans, was enabled by jumper connection. Since an external scan initiation was desired, pin 34 on the ST-800 J2 connector was grounded. Effectively, the scan-clock option circuitry was used to initiate each scan. The actual signal came not from the scan clock, however, but instead from a negative TTL pulse which was input at pin 36 of the ST-800 J2 connector from an external pulse generator.

4. Direct Memory Access Interface

The Intel Direct Memory Access controller board was installed in the MDS-800 mainframe and connected to the ST-800 converter. The DMA was set to operate at interrupt level four by adjusting a rotary switch on the board. Upon

completion of a cycle, the DMA generated a signal to the CPU interrupt controller which then stopped program execution in order to service the interrupt.

III. SOFTWARE

All programming on the project was done using options available under the CP/M (Control Program/Monitor) monitor. This operating system allows the user to manage files on disk and provides the basic input/output facilities necessary to communicate with peripheral devices. System utilities allow the user to create, edit, load, run, and record programs on the diskette. Two powerful programs, the Macro Assembler (MAC) and the Symbolic Instruction Debugger (SID), give the user vast capabilities to assemble and monitor programs at execution in order to easily detect errors. The system is a product of Digital Research and is described in Ref. 10.

The programming language options available were assembly language and PL/M. Assembly language is shorthand notation for machine language which allows mnemonic instructions, with a one-to-one correspondence between each assembly instruction and a machine code instruction. Because of this, assembly affords direct control over the working registers of the central processor unit; however, for the same reason even simple jobs for the microprocessor can result in long and complex programs. Programs must first be assembled, whereby the assembly mnemonics are compiled into

hex code and addresses are assigned to symbols. Next the program must be loaded, or converted to binary code, before execution by the microprocessor. The only alternative is the PL/M language which is somewhat more sophisticated but which, when reduced finally to binary code, results in about a twenty-five percent waste of memory. The 8080 assembly language was therefore used in all programs during this project.

All assembly programs devised for this project required the use of large memory buffers, so efficiency of programming was paramount in order to reserve as much memory space as possible. For the commonly used 48K system, for example, the memory locations 0-100H and A900H-BFFFH were used for the operating system code. If the user program occupied storage locations 100H-1000H, only 39,078 locations (A900H-1000H) remained available for data storage.

Another observed disadvantage of the assembly language was that the programs were difficult to follow even when well-documented. For this reason, all the programs were designed to be "user oriented" with a multitude of prompts and explanatory comments being echoed to the CRT. Additionally, the programs were heavily documented and instruction guides written for each interface.

The software which interfaced the analog to digital converter, high-speed line, and printer to the microprocessor could all be classified as monitor and control programs. The peripheral devices were monitored and controlled by the central processor unit while keeping the user informed via the CRT.

A. PRINT Program

The PRINT program searches the disk for a specified file, loads the file into memory buffer, and outputs the file to the Model 40 Printer. While outputting the file, PRINT also creates a format for the standard 11x14 inch paper, numbers the pages, and heads each page of printout with the given filename and filetype. If desired by the user, the PRINT program will double space the output; this option works in conjunction with the single/double space switch inside the printer cabinet. PRINT is compatible with all ASCII filetypes.

Another option allows the partial printout of a file between two specified strings of data. This feature is especially useful when working with large files and conserves both paper and time.

Most source files residing on the user's disk are not pre-formatted, hence the PRINT program produces a neat,

orderly output with numbered and titled pages. Certain files, however, including PRN files generated by the Macro Assembler or the Tex Formatter, have already been formatted for a similar output. In order to avoid double formatting, an option exists in the PRINT program whereby the user is queried whether the named file is already formatted. An affirmative response causes the formatting and page numbering features of the program to be suppressed.

1. Printer Control

The program's first task is to initialize the printer and to output data at a rate commensurate with the printer's ability. The CPU first sets up the appropriate timer on the SBC 534 board to pace the binary output at 9600 bits per second. Next the USART is commanded to transmit seven bit words (the eighth bit is zero for all ASCII characters) with one start bit, one stop bit, and no parity bit. The entire serial word train involves ten bits of data. Additionally the CPU command resets any USART error flags and drives the DTR line high, thus turning on the printer motor.

Once the USART is initialized, the CPU reads its status and checks the condition of the Transmitter Empty (TxE) flag. As soon as the transmitter buffer is determined to be empty, the CPU outputs the next data byte.

2. File Reading

Using CP/M system functions, the file to be printed is found and read from the diskette. Since the CP/M disk read function reads 128 byte blocks of data at once, another CP/M function is used to increment the memory location by 128 for each block of data read from the diskette. This process continues until the byte "1AH" is encountered signifying the end of file (EOF).

3. Formatting

Counters are maintained to limit each line to 131 characters and each page to 55 lines. At the beginning of each page the page number, filename, and filetype are output. At the end of each line the keyboard is checked for a user interrupt. The process continues until the end of file (EOF) byte is again encountered. At this time the program turns off the printer motor and returns to the CP/M environment.

4. Prompts

Once the program is executed, user prompts flow sequentially to the CRT and the responses are checked for reasonableness. Any problems associated with incorrect responses, file reading, or control of the printer result in

automatic error messages to the console.

5. PRINT User's Guide

The PRINT User's Guide was intended to be used as an independent manual. The guide provides detailed operating instructions for the Model 40 Printer interface and is included as Appendix E. A listing of the PRINT Assembly program is included as Appendix I.

B. LINK Program

Programming for the high-speed line interface was difficult because the absence of handshaking on the line presented some unique problems. When transmitting from the microprocessor to the IBM 360, the rate and regularity at which data words were output were of no significance. The IBM 2701 unit received one complete line before answering. Upon receiving a byte "13H" (XOFF) signalling the end of a line, the 2701 unit answered with a sequence of bytes: "0DH" (carriage return), "0AH" (line feed), "00H" (null), "3EH" (CMS prompt ">"), and "11H" (XON). Any information transmitted by the IBM 360 always preceeded this exact sequence. The programmed arrangement was, therefore, that each unit would take turns transmitting and receiving.

More complicated provisions had to be inserted into the program, however. If the microprocessor attempted to transmit a line containing more than 132 characters, the 2701 unit rejected the excess characters and interrupted with an error message. Also there were occasional instances when the IBM 360 output a large number of lines without the XON. For example, if commanded to print a FORTRAN file, the IBM 360 would output the entire file before transmitting the XON. Therefore, the capability of interrupting the IBM 360 was needed. Instead, the control program had to allow for reception while transmitting and for transmission while receiving.

This was accomplished by setting up two separate loops for the transmit and receive functions. When involved in the reception of characters, the microprocessor CPU constantly checked the keyboard for a user interrupt. If one were found, the program immediately issued a pair of XON characters to the 271 unit while still receiving characters. When the 2701 received the XON´s, it acknowledged the interrupt with the usual sequence.

When involved in the transmission of characters, the CPU constantly checked the receive buffer for a data word. When one was found, the program control reverted to the receive function.

1. USART Setup

The USART and timer for the high-speed line were set up similarly to the printer USART. The timer was commanded to generate a baud rate of 1200 baud and the USART was commanded to both transmit and receive. The transmitted serial word train contained one start bit, seven data bits, and two stop bits. The only available baud rate on the high-speed line was 1200 baud. Future improvements to the rate are discussed in the conclusion section to this thesis.

2. Monitor Function

When executed, the LINK program was in the receive status. After receiving the first transmission from the IBM 360, program control went into the transmit function. While in this status, the CPU program alternated between checking the receive buffer for an interrupt and checking the keyboard for a user input. Upon receipt of a user input, the CPU screened the input for certain control characters and, if one were found, branched to the proper subroutine. This monitor function was designed so that control characters used during CP/M operation could also be used when operating with the IBM 360 under CMS. User inputs that were not control characters were output to the IBM 360.

A Control I, the tab command under CP/M, was transmitted to the IBM 360 as a "?" which should have been previously defined to CMS as a logical tab character. A RUBOUT was transmitted as a CMS delete character symbol and a Control U as a delete line symbol. A Control R or Control T caused program control to branch to subprograms that effected the transfer of complete files between microprocessor diskette and IBM disk. Similarly, a Control P caused control to branch to a routine that turned on the printer if off and vice versa. This allowed the user the capability of echoing all correspondence with the IBM 360 to the printer.

If a Control C were input, the program control instituted a soft boot and returned the user to the CP/M environment. The high-speed line was still active although the LINK program was no longer in service. Any transmissions by the IBM 360 at this time "fell on deaf ears". A Control G caused the program to print on the console a list of all Control functions.

3. Data Buffers

Although the high-speed line operating at a baud rate of 1200 baud was usually slower than the microprocessor and all its peripherals, there was one circumstance when the LINK program could not keep pace with the line. If the

printer option were on and a line feed character were being implemented, a delay resulted while waiting for the printer to get ready for the next character. To provide for this circumstance, all data received from the IBM 360 was routed through a First-In-First-Out (FIFO) buffer. After determining that the USART receive buffer did not have a byte ready, the CPU next checked both the CRT and printer to determine if they were ready to receive a byte. If so, the last byte received was output. If either the CRT or printer were not ready, the byte was stored in the FIFO buffer and the USART receive buffer rechecked. In practice the buffer usually expanded after encountering a line feed character because of the printer delay, but caught up before the end of the next line due to the superior baud rates of the CRT (2400) and the printer (9600).

Another type of buffer was utilized in the transmit file and receive file subprograms. A file to be transmitted to the IBM 360 was first completely loaded into memory before transmission, similar to the operation of the PRINT program. If the file size exceeded the available memory, then part of the file was loaded and transmitted, and then another part until the end of the file was encountered. For the 48K system the memory available as a data buffer was about 38K. For files being received from the IBM 360, an insurmountable problem sometimes arose. The file was being received too fast to simultaneously write on the diskette,

so the data had to be buffered. If the file exceeded the available memory, then transmission by the IBM 360 had to be stopped immediately to avoid losing any of the file. Because of the timesharing operation of the IBM 360 under CMS, the transmission could not be immediately interrupted. Since this anomaly could not be corrected, it was determined that the user would have to limit incoming files to 38K or else break up larger files into 38K segments.

4. LINK User's Guide

Precise instructions for the operation of the LINK program are contained in the LINK User's Guide, Appendix C. The assembly program listing is included as Appendix G.

C. GO Program

The GO program controls the operation of the ST-800 Analog to Digital Converter with the Direct Memory Access Controller. The primary concern in designing this system was to effect the fastest possible data sampling rate while maintaining a high degree of flexibility. The crucial element of speed and the complexity of the component interaction combined to make the software development for this system quite a challenge.

When operating with the DMA, the ST-800 does not communicate directly with the CPU. The DMA is programmed with the total number of converted data bytes to be passed and the memory address at which to store the first byte. The ST-800 is programmed through the DMA with regard to the initial and final channels to be converted. The process of converting the analog signal inputs for the initial through final channels and passing them to the DMA is known as a scan. Full handshaking between the DMA and ST-800 circuits is employed and the throughput time for converting an analog signal into two hex bytes and passing both bytes through the DMA to random access memory is approximately twenty-two microseconds. When one scan is completed, the ST-800 relies on either the CPU or a signal from the scan clock to initiate another scan. When the word length register in the DMA counts down to zero, the DMA has finished its programmed task and waits to be reset.

Initially the approach toward meeting the primary goal was to set up the system on a dual-interrupt basis. Although this scheme provided tremendous flexibility, in some cases it retarded the conversion process from full speed operation. Another configuration was ultimately adopted, but the dual-interrupt approach had some merit and is discussed under the heading of Alternative Solutions.

The Scan-clock Option on the ST-800 provides for initiation of subsequent scans after the first is completed. An end-of-scan signal starts a preset countdown clock which, when timed out, initiates the next scan. The disadvantages to this option were that hardware changes were required to vary the countdown interval, and the fastest scan repetition rate was 1000 scans per second.

By enabling the Scan-clock Option but disabling the countdown timer itself, an external pulse could be applied to initiate scans through the scan-clock circuitry. This method was adopted as the most flexible as well as the fastest.

1. Data File Parameters

The contents of a data file is a collection of hex digits and two such files would be indistinguishible without additional information. The first file of data was named DATA01.XXX and subsequent filenames were incremented by one digit. Through a sequence of user prompts and responses, the program determined which options the user desired. This information was used to set up the data conversion run and also was recorded in the data file to facilitate later identification. Included in the file information block were the initial and final channels, number of data points in the sample, scan repetition rate, run coordination number, and

the number of data bytes involved in each scan.

2. ST-800 and DMA Setup

The number of data points specified by the user was multiplied by two since each digitized data word required two bytes of storage. The result was programmed into the word length register of the DMA. The initial and final channels to be scanned were loaded into the ST-800 via the DMA. The memory location 900H was programmed into the DMA as the future address of the first converted data byte. The DMA controller was then commanded to transfer data from the ST-800 to memory. The ST-800 was commanded by the CPU to start conversion.

3. DMA Reset

Since the pulse generator which initiated subsequent scans was disabled at this point in time, the ST-800 converted through one complete scan and stopped. The word length register on the DMA was not decremented to zero after one scan, hence no interrupt was forthcoming. This first dummy scan was necessary simply to synchronize the ST-800 with the pulse generator.

The word length register and memory address register were now reloaded with their initial values. The DMA was

given a new command word which allowed it complete control of the data bus and the user prompted to enable the pulse generator. By this method the first data byte from the first channel went into the first memory location. The channels were converted at the maximum throughput rate of the ST-800-DMA combination (about 45,000 Hertz) until each scan was completed, and the scan repetition rate coincided with the pulse generator output. When the entire data sample was finished, the word length register decremented to zero and the DMA issued a level four interrupt. A jump vector which had been previously inserted into the RST 04 location directed program control to a routine which serviced the interrupt, disabled the DMA, and prompted the user to disable the pulse generator. Lastly the program wrote the data file to the system diskette if desired by the user and then set up for another run.

4. GO User's Guide

The GO User's Guide, Appendix B, provides the details for setup and operation of the data acquisition system. The GO Assembly program is listed in Appendix F.

D. DATLINK Program

The DATLINK program is a modification of LINK and is identical in most respects. Since the data acquired with

the GO system was recorded on the diskette in hex bytes, each byte had to be converted into two ASCII characters before transmission over the high-speed line. The transmit file mode of DATLINK limited each line to the number of data bytes obtained from each scan. Therefore files created under CMS on the IBM 360 were already formatted with one scan per line.

Because of the additional code needed to accommodate the data files, the receive file mode was removed from the DATLINK program. The User's Guide for DATLINK is included as Appendix D and the Assembly program listing is Appendix H.

IV. SYSTEM QUALIFICATION

System qualification was achieved by digitizing known analog signals, storing the data files on diskette, and transmitting the files to the IBM 360 for data reduction. The output files were then transmitted back to the micro-computer system, stored on diskette, and output to the line printer.

A. Shannon's Sampling Theorem

When digitizing a signal, care must be taken to ensure that Shannon's Sampling Theorem is obeyed; otherwise there is a possibility of aliassing occurring. In general, a degree of conservatism should be followed when digitizing such that ten to fifteen samplings should take place each fundamental period and at least ten to fifteen waveforms should be recorded. If the presence of higher harmonics were suspected, added conservatism should be used.

B. Qualification Test

Sinusoid waveforms with carefully measured frequencies of 20, 200, and 1000 Hertz were chosen for data sampling. The system was set up according to the GO User's Guide,

Appendix B, and the scan triggering pulse generator frequency was measured at 300, 3000, and 10,000 Hertz, respectively. After the data was acquired and stored, the files were sent via the DATLINK program to the IBM 360. Next, using the LINK program, a FORTRAN reduction program was created within the IBM computer similar to the BASIC program reported by Pickelsimer, Ref. 13, and Englehardt, Ref. 1.

C. Data Sampling Theory

One common form of unsteady data recording involves periodic natural signals of arbitrary waveform having a well-established fundamental frequency. As an example, instrumentation transducer system transfer functions would involve data records at various prescribed frequencies of input and output signals. The systems described in this thesis are naturally oriented for providing transfer function type of information using the following cross-correlation scheme to pick out the Fourier components of a deterministic type waveform. Consider a data set X(1), X(2), X(3), ... ,X(N) representing a waveform of a known frequency which has been sampled at given intervals. After truncating the set to an integral number of periods, the bias or average value can be determined and removed from each member of the set.

D. Fourier Analysis

Any periodic waveform can be represented by the Fourier Series

$$X(t) = \sum_{n=1}^{\infty} [A_n \cos n\omega_1 t + B_n \sin n\omega_1 t + A_o]$$

and the coefficients can be found by

$$A_o = 1/T \int_0^T X(t)\, dt$$

$$A_n = 2/T \int_0^T X(t) \cos n\omega_1 t\, dt$$

$$B_n = 2/T \int_0^T X(t) \sin n\omega_1 t\, dt$$

In cases where the data set represents a known simple waveform (no harmonics) such as the sinusoid used in the system qualification, the Fourier coefficients can be obtained by an estimation procedure. For the assumed truncated data set with bias removed

$$Y(1),\ Y(2),\ Y(3),\ \ldots, Y(M)$$

representing a discretized sinusoid signal with frequency F and scan rate of ΔT, the first harmonic estimates become

$$A = (2/M) \sum_{I=1}^{M} Y(I) \cos [2 \text{ Pi } F \Delta T\ (I)]$$

$$B = (2/M) \sum_{I=1}^{M} Y(I) \sin [2 \text{ Pi } F \Delta T\ (I)]$$

and the magnitude and phase are estimated by

$$C = [\, A^2 + B^2 \,]^{1/2}$$

$$\phi = \text{Tangent Inverse } [\, -B/A \,]$$

Higher harmonics, such as the Kth, can be estimated by replacing [2 Pi F ΔT (I)] with [2 Pi (K) F ΔT (I)] in the above equations.

Had the data set X(1), ...,X(N) resulted from a random waveform, the above formulae conceptually would be replaced by applying a Fast Fourier Transform algorithm to the data set. This procedure is built into several existing programs in the Computer Center library.

E. Interchannel Sampling Delay

The Fourier Coefficient estimation procedure described above was used during system qualification to establish the interchannel sampling delay. The scan rate or sampling rate refers to the time involved between converting the (Ith) and (Ith + 1) samples of a specific input channel. This scan rate is adjustable since it is controlled by an external pulse generator serving as a trigger. Whenever more than one channel is being digitized, there is a slight time difference between the instants of sampling for the respective channels. This time difference is known as the interchannel sampling delay and is not adjustable since it is established

by the throughput rate of the Analog to Digital converter-DMA controller combination.

F. REDUCE Fourier Coefficient Program

The FORTRAN program created to reduce the system qualification data was similar to the BASIC program used by Englehardt in Ref. 1. Since the test signals were simple waveforms with known frequencies, the estimation procedure described above was used. The REDUCE Fortran Program, listed as Appendix M, was written to accommodate data from four input signals. Since the same test signal was applied to each of the four input channels, the phase differences evident in the reduced data sets gave a close determination of the interchannel sampling delay (21.7 microseconds).

G. System Qualification Results

The reduced data from the three test runs are presented in Appendix N. The sinusoid waveforms had identical magnitudes and that fact was reflected on all four channels of data for each of the three test runs. The magnitudes of the second harmonics were approximately 0.3 percent of the first harmonic magnitude in each case. The existence of a second harmonic was attributable to slight imperfections in the sinusoid generator used for the test waveforms.

The most significant finding from the reduced data was the interchannel sampling delay. For each test run, the difference in phase between two consecutive channels, when divided by the period of the test waveform, indicated a delay of approximately 21.7 microseconds. The throughput rate for the combination of Analog-to-Digital converter and DMA controller was faster than had been predicted. Therefore the maximum sampling rate of the data acquisition system was determined to be slightly in excess of 45,000 Hertz, as compared to the initial value of 40,000 Hertz estimated.

V. ALTERNATIVE SOLUTIONS

The existence of multiple solutions to a specific problem leads to a variety of approaches in microcomputer application. Hardware selection between commercially manufactured or user-constructed devices, the choice of hardware or software to accomplish a given task, and the infinite approaches of software itself exemplify some of the decisions facing the potential user.

Initially a circuit board was constructed for the purpose of driving the Model 40 Printer and high-speed line. Many design problems were encountered and valuable experience was gained. However, the Intel SBC 534 Input/Output Board was later utilized because of its capacity for future system improvement.

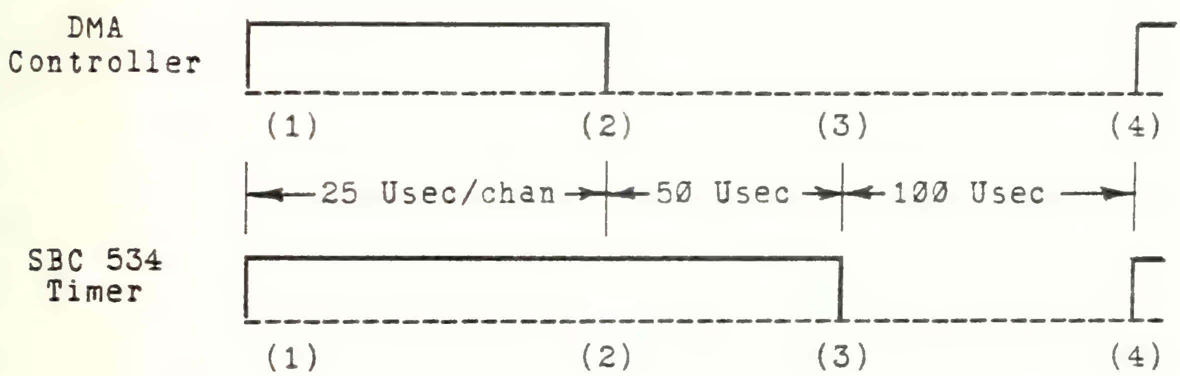
A. Dual-Interrupt Data Acquisition

The concept first implemented in setting up the Analog to Digital Converter and the Direct Memory Access controller was to use a timer circuit contained on the SBC 534 board to initiate each scan. A jumper selectable option on the SBC 534 permitted the series operation of two timers. One timer served as a clock for the second timer which initiated an

interrupt signal after counting down to zero. The DMA controller and SBC 534 board were hardwired to generate level four and level five interrupts, respectively. The DMA controller and Analog to Digital Converter were programmed for one complete scan followed by an interrupt. The timers and interrupt controller on the SBC 534 board were programmed to delay for a specific interval before interrupting. Starting both processes together, the program waited for the DMA controller interrupt indicating the end of the scan, and then reset the DMA controller. When the timed interrupt occurred, a software routine reset the timers and re-initiated the two circuits. When the desired number of data points had been converted, the program disabled the interrupt mechanism and wrote the data on the system diskette.

While the operational details of the dual-interrupt setup are contained in the G02 program listing, Appendix J, this approach was ultimately replaced by the system already described. Two substantial obstacles to its successful operation were never overcome. The presence of the SBC 534 board installed in the MDS mainframe caused a level five interrupt during the bootstrap operation resulting in an aborted disk drive interface. A patch inserted into the CP/M BIOS program averted the untimely interrupts, but a more significant problem remained.

The interrupt service routines were long and cumbersome, particularly the routine that reset the SBC 534 timers. In order to effect the exact desired interval between scans, the time required to implement the reset instructions was taken into account by modifying the countdown interval to a value of 100 microseconds less than the scan interval. This difference was estimated by totalling the instruction cycle times in the routine. Also, the DMA interrupt service routine had to be completed before the timer interrupt occurred so as to avoid stacked interrupts. As shown in Fig. 3, the allowable conversion time of approximately twenty-two microseconds per channel (1 - 2) was 150 microseconds less than the scan period.



(1) Conversion process and interval countdown started
(2) Conversion ended - interrupt 4 - service routine begins
(3) DMA reset complete - timed interval ends - interrupt 5 - interval reset begins
(4) Timer reset complete - new scan begins

Figure 3 - Dual Interrupt Timing

These software delays resulted in a considerable limitation on the maximum scan rates. With the dual-interrupt process,

the eight channel scan rate was only 2800 Hertz and the one channel rate was 5000 Hertz. With the externally timed system described earlier, the eight channel and one channel scan rates were 5000 Hertz and 45,000 Hertz, respectively.

B. Model 40 Printer as a List Device

The CP/M system provides for the operation of a "list" device which originally was designated as the teletype terminal. Several routines within the BIOS program and the MDS monitor divert the microprocessor output to the list device. For example, the CP/M routines TYPE and PIP, as well as the monitor function LO (for List Out), are directed to the list facility. Additionally, by depressing a Control P key, the user can cause all characters directed to the console to also be echoed to the list device. Before the printer can be used as the CP/M list device, it must be initialized by a separate routine such as the ON Assembly program which is included as Appendix L, and the CP/M itself must be altered to address the printer.

A simple patch to the CP/M BIOS program, included as Appendix K, can be used to alter the system so that output to the list device can be redirected to the Model 40 Printer. If the printer USART were programmed beforehand to accept data, the patched CP/M could produce a printed copy of all the information presented on the console. The patch

may be implemented under DDT control and the patched file can be used to generate a patched system disk.

VI. CONCLUSIONS

The data acquisition system developed during this project provided an extremely flexible, dynamic tool for investigating rapidly changing experimental aerodynamic phenomena. Signals from analog measuring devices were sampled at a maximum rate of 45,000 times per second and the data stored on magnetic disks. The data was then expeditiously transferred to the IBM 360 computer where higher level language programs directed the efficient reduction of raw data to formatted answers. The empirical results were then returned to local microprocessor environment and printed. The printer was operated alone to produce hard copy source listings, records of microprocessor functions, and text formatted printouts such as this document.

A. Future System Improvements

The speed at which data files were transmitted to the IBM 360 computer was limited by the IBM 2701 Data Adapter unit to 1200 baud or about 120 characters per second. Although the rate increase over earlier interfaces was by a factor of eleven to one, the capability exists to further improve the speed another eight times to a rate of 9600 bits per second. The MDS system including hardware and software

was designed to run at the higher speed and only minimal software changes would be necessary to effect such an improvement. Because other users cannot accommodate the 9600 baud, the IBM 2701 unit is hardwired to operate at only 1200 baud.

The scheduled expansion of the IBM interface for high speed line operation will provide a line hardwired to operate at 4800 baud. Whenever the IBM facilities are modified, the microprocessor can be upgraded by making some minor changes to the LINK and DATLINK programs. The countdown number applied to the high-speed line USART should be altered in both programs to generate the faster baud rate. Also, during operation under the receive file mode of the LINK program, a subroutine "CONOUT" echoes all received characters to the CRT terminal. Since the CRT baud rate of 2400 baud is less than 4800, the instruction "CALL CONOUT" (08B8H) should be deleted.

APPENDIX A

Glossary

ASCII: American Standard Code for Information Interchange. This is a seven-bit-plus-parity code established by the American National Standards Institute to achieve compatibility between data services.

assembler: a compiler that translates assembly language into hex code and assigns memory locations to labels.

assembly language: programming language used in microcomputer applications.

baud: a serial data transmission rate expressed in bits per second.

BIOS: Basic Input/Output Operating System - a subprogram of the CP/M system that effects all transfers of information between the CPU and its peripheral devices.

bit: binary digit - a single unit of information in a binary word.

buffer: a block of random access memory that has been reserved for temporary data storage.

byte: an eight-bit binary word which is processed as a single quantity.

CMS: Cambridge Monitoring System - a time sharing scheme used by the IBM 360 computer which allows several users simultaneous access to a single virtual machine.

CRT: cathode ray tube - a television-like picture tube used in visual display terminals.

CP/M: Control Program/Monitor - a software system which allows the microprocessor to be operated as a microcomputer. The system is described in Ref. 10.

CPU: Central Processor Unit - the area of the microcomputer

that computes and controls all logical and arithmetic functions.

DMA: Direct Memory Access - a facility whereby input/output data can be transferred to/from memory without passing through the CPU.

FIFO: First-In-First-Out - a buffer in which data is inserted and removed in the same order.

hardware: the physical circuitry and related devices within the microprocessor.

Hertz: units of rate of repetition (cycles per second).

hex: number system based on 16 decimal - one hex digit equates to four binary bits; e.g., 14 decimal is E hex or 1110 binary.

instruction cycle: a finite time span during which the CPU executes programmed instructions. For the MDS this time span can be as short as 2 microseconds. The instruction cycle time may be computed by multiplying the number of clock cycles in a given instruction by 0.5 microseconds.

interrupt: an independent circuit and logic system within the microcomputer. Certain peripheral devices can signal the interrupt logic controller which screens interrupt priorities so that several simultaneous signals can be processed. The interrupt controller halts program execution and diverts the CPU's attention to a subroutine that services the interrupt.

K: symbol used to denote one kilo-byte (1024 decimal or 400 hex bytes) of memory.

machine code: the bit patterns actually used by the CPU to execute its assigned logic functions.

MDS: Microcomputer Development System - the Central Processor Unit with related memory and peripheral devices.

peripheral device: any major independent component controlled by the CPU; e.g., the CRT, teletype, printer, disk drive, or Analog to Digital Converter.

PL/M: Programming Language/Medium.

RAM: random access memory - volatile memory area used for program code and data storage.

RS-232C driver: a transistorized switching device which converts TTL voltage levels to +/- 15 volts for longer range transmission. The RS-232C refers to an Electronic Industries Association (EIA) specification for the device.

ROM: Read Only Memory - non-volatile memory in a computer which contains permanent machine code.

software: the program which contains routines to operate the microcomputer.

throughput: refers to the elapsed time for one complete cycle; e.g., the Analog to Digital Converter throughput includes the time to sample and convert an input, pass the digitized word to the DMA, and set up for the next cycle.

TTL: Transistor Transistor Logic - low current logic devices operate with five volts D. C. power supplies. Subsequently a logical true state is indicated by +5 volts and a false state by 0 volts.

Usec: microsecond - one millionth of a second.

USART: Universal Synchronous Asynchronous Receiver Transmitter - integrated circuit device which converts parallel transmissions into serial transmissions and vice versa.

XON: an ASCII "11" which signifies the beginning of a transmission.

XOFF: an ASCII "13" which signifies the end of a transmission.

APPENDIX B

GO USER'S GUIDE

I. CAPABILITIES

A. GO INTERFACES THE INTEL MDS 800 MICROPROCESSOR AND DIRECT MEMORY ACCESS CONTROLLER BOARD WITH THE DATEL ST-800 ANALOG TO DIGITAL CONVERTER BOARD FOR HIGH SPEED DATA ACQUISITION. A MAXIMUM OF 16 CHANNELS OF ANALOG DATA CAN BE INPUT, CONVERTED, AND STORED IN RANDOM ACCESS MEMORY AT A RATE OF 45 KHZ.

B. GO INTERFACES A SEQUENCE OF PROMPTS AND USER RESPONSES. THESE RESPONSES ARE USED BY THE PROGRAM TO SET UP THE ANALOG TO DIGITAL CONVERTER AND DIRECT MEMORY ACCESS CONTROLLER TO PROVIDE A LEVEL FOUR INTERRUPT WHEN DATA HAS BEEN ACQUIRED.

C. GO WRITES EACH BLOCK OF ACQUIRED DATA ONTO A FLOPPY DISK FOR LATER RETRIEVAL. EACH DATA FILE CONTAINS FORMATTED PARAMETERS WHICH DESCRIBE THE DATA SAMPLING PROCEDURES, SUCH AS NUMBER OF DATA POINTS, SCAN RATE, AND A RUN COORDINATION NUMBER WHICH IS ENTERED BY THE USER.

D. A VARIABLE FREQUENCY PULSE GENERATOR IS USED DURING THE DATA ACQUISITION PROCESS TO INITIATE EACH SCAN. CARE MUST BE TAKEN TO AVOID SELECTING A SCAN RATE WHICH EXCEEDS THE SYSTEMS CAPABILITY. FIGURING A THROUGHPUT TIME OF TWENTY-TWO MICROSECONDS PER CHANNEL FOR CONVERSION TO MEMORY STORAGE, THE SELECTED PULSE RATE SHOULD NOT EXCEED 45,000 DIVIDED BY THE NUMBER OF CHANNELS; E.G., IF EIGHT CHANNELS WERE TO BE SAMPLED, THE SCAN RATE SHOULD NOT EXCEED 5500 SCANS PER SECOND.

E. SUCCESSIVE DATA SAMPLING RUNS ARE RECORDED ON THE FLOPPY DISK IN DRIVE B WITH FILENAMES DATA01.XXX, DATA02.XXX, ETC. IF A LIKE FILENAME ALREADY EXISTS ON THE DISK, IT IS DELETED BEFORE THE NEW FILE IS WRITTEN.

II. SETUP

A. ANALOG INPUTS ARE LIMITED TO PLUS OR MINUS FIVE VOLTS AND SHOULD BE CONNECTED TO THE SYSTEM THROUGH A LOCALLY CONSTRUCTED INPUT TERMINAL. THE ANALOG TO DIGITAL CONVERTER CAN THEN BE CALIBRATED BY EXECUTING A DATEL TEST PROGRAM ST-800 (AVAILABLE ON DISK AND PAPER

TAPE IN THE MICROPROCESSOR LAB).

B. A NEGATIVE TTL PULSE (WHICH STROBES ZERO VOLTS) IS ALSO CONNECTED TO THE INPUT TERMINAL. A DIGITAL FREQUENCY COUNTER SHOULD BE INTERCONNECTED TO OBTAIN PRECISE SCAN RATE INFORMATION. THE PULSE GENERATOR SHOULD BE TESTED AND THEN PLACED IN A STANDBY CONDITION (NO PULSING).

C. A PREFERABLY BLANK, FORMATTED DISKETTE SHOULD BE PLACED IN DISK DRIVE B.

III. OPERATION

THE GO PROGRAM IS EXECUTED BY THE FOLLOWING COMMAND:

GO <CARRIAGE RETURN>

IMMEDIATELY THE USER IS PROMPTED WITH

ENTER STARTING CHANNEL

FOLLOWING USER'S REPLY, THE NEXT PROMPT APPEARS:

ENTER FINAL CHANNEL

NOTE: RESPONSE TO THE ABOVE TWO PROMPTS SHOULD BE IN THE RANGE OF Ø - 15. IF THIS RANGE IS EXCEEDED OR IF THE STARTING CHANNEL IS GREATER THAN THE FINAL CHANNEL, ANOTHER PROMPT APPEARS:

TRY AGAIN, TURKEY

AND THE ABOVE PROMPTS ARE REPEATED.

NEXT THE USER IS PROMPTED WITH A CHOICE OF DATA BLOCK SIZES:

ENTER DESIRED NUMBER OF DATA POINTS

ENTER	DATA POINTS	DISK SPACE
A	1Ø24	2K
B	4Ø96	8K
C	1Ø24Ø	2ØK
D	2Ø48Ø	4ØK
E	26624	52K(62K SYSTEM)

THE USER SELECTS ONE OF THE OPTIONS BY TYPING THE APPROPRIATE LETTER AND A CARRIAGE RETURN.

USER IS THEN PROMPTED WITH

ENTER SCAN RATE

THIS RESPONSE CAN BE ENTERED IN ANY FORMAT

NOTE: THE ACTUAL SCAN RATE IS DETERMINED BY THE PULSE GENERATOR. THE RESPONSE TO THE ABOVE PROMPT WILL APPEAR IN THE FILE INFORMATION PARAMETERS.

THE NEXT PROMPT IS

    ENTER COORDINATION NUMBER

THIS RESPONSE CAN BE ANYTHING THE USER MIGHT CHOOSE TO DISCRIMINATE BETWEEN VARIOUS RUNS.

FINALLY THE SYSTEM INDICATES A READY CONDITION BY

    START PULSE GENERATOR

AT THIS TIME OR WHENEVER USER CHOOSES, THE PULSE GENERATOR SHOULD BE CHANGED FROM A STANDBY TO PULSING CONDITION. THE COMPLETION OF A RUN IS SIGNALLED BY A BEEP AND

    RUN COMPLETE - DISABLE PULSE

THE PULSE GENERATOR SHOULD BE RETURNED TO A STANDBY CONDITION AT THIS TIME. THE USER IS PROMPTED WITH

    WRITE DATA FILE ON DISK?? (Y/N)

IF USER SELECTS ANY KEY BUT "N", THE PROGRAM WILL ECHO THE FILE PARAMETERS TO THE CONSOLE FOR USER VERIFICATION AND WRITE THE DATA FILE ONTO THE DISKETTE IN DRIVE B. ANY PROBLEM INCURRED IN THE WRITE PROCESS WILL BE DETAILED BY EITHER

    DISK WRITE ERROR - TRY ANOTHER

        OR

    DISK FULL

AFTER PLACING A CLEAN DISK IN DRIVE B, USER SHOULD TYPE A CARRIAGE RETURN TO START THE WRITE PROCESS AGAIN.

NOTE: REGARDLESS WHETHER THE DATA ACQUIRED IN A RUN IS WRITTEN ON A DISK, THE DATA FILENAME WILL BE INCREMENTED.

THE NEXT PROMPT TO APPEAR IS

    ANOTHER DATA RUN DESIRED?? (Y/N)

SELECTION OF Y WILL START THE PROMPTS AGAIN, AND SELECTION OF ANY OTHER KEY WILL REBOOT THE SYSTEM AND

RETURN USER TO CPM.

NOTE: IF THE PROGRAM IS NOW RE-EXECUTED, THE DATA FILENAME COUNT WILL START OVER AT DATA01.XXX AND OVERWRITE PREVIOUS DATA FILES.

IV. DATA FILES

AN ACQUIRED DATA FILE CAN BE DUMPED UNDER CP/M. THE FIRST 128 BYTE BLOCK OF THE FILE CONTAINS INFORMATION RELATING TO ITS ACQUISITION. A SAMPLE DUMPED FILE FOLLOWS:

```
44 41 54 41 30 31 01 07 31 30 32 34 24 35 30 30
30 24 30 30 39 31 31 30 30 33 24 00 00 00 00 00
12 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
10 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
00 00 00 ETC
```

THE FIRST SIX BYTES ARE THE FILENAME IN ASCII
DATA01
THE NEXT TWO BYTES ARE THE INITIAL AND FINAL CHANNELS IN HEX
01,07
THE NEXT THREE PARAMETERS ARE ASCII CODE INDICATING THE NUMBER OF DATA POINTS, SCAN RATE, AND RUN COORDINATION NUMBER, EACH FOLLOWED BY THE DELIMITER "$"
1024
5000
00911003
THE '12' INDICATES THAT 1200H WAS THE UPPER LIMIT ON MEMORY USED -
THE '10' IS THE HEX REPRESENTATION OF THE NUMBER OF MEMORY BYTES PER SCAN
THE REMAINDER OF THE BLOCK IS ZEROES

M. T. ELLIOTT, NPGS
AUGUST 28, 1978

APPENDIX C

LINK USERS GUIDE

I. LINK INTERFACES THE MDS 800 (AND MODEL 40 PRINTER) WITH CP/CMS THROUGH A 1200 BAUD TELEPHONE LINE. BOTH THE LINE AND THE PRINTER ARE DRIVEN BY 8251 USARTS INCORPORATED IN AN SBC534 I/O BOARD. LINK OPERATES IN ONE OF THREE MODES AS FOLLOWS:

A. DIRECT LINKUP MODE

1. TRANSMITS CHARACTERS TYPED ON KEYBOARD TO CP/CMS WITH SOME FILTERING BUT NO BUFFERING; ECHOES CHARACTERS TO CONSOLE (AND PRINTER)

2. RECEIVES CHARACTERS FROM CP/CMS AND UTILIZES A FIFO BUFFER TO PRINT THE CHARACTERS ON THE CONSOLE (AND PRINTER)

3. ALTHOUGH NO HANDSHAKING IS UTILIZED ON THE LINE, SOFTWARE PROVISIONS ALLOW EITHER END TO INTERRUPT THE OTHER'S TRANSMISSIONS

4. CERTAIN CHARACTERS TYPED ON THE KEYBOARD ARE FILTERED OUT:

RUBOUT	- BACKSPACES THE CONSOLE AND TRANSMITS A DELETE CHARACTER SYMBOL "@"
CONTROL I	- ECHOES AND TRANSMITS A "?" TO INDICATE A LOGICAL TAB - NOTE: "?" MUST BE PREVIOUSLY DEFINED TO THE CMS AS A TAB CHARACTER
CONTROL U	- TRANSMITS A DELETE LINE SYMBOL "["
CARRIAGE RETURN	- TRANSMITS END OF LINE SYMBOL AND WAITS FOR AN ANSWER

ADDITIONAL CONTROL CHARACTERS ALTER PROGRAM EXECUTION AS FOLLOWS:

CONTROL C - REBOOTS SYSTEM

CONTROL D - RETURNS USER TO DIRECT LINKUP MODE

CONTROL P - TURNS PRINTER ON IF OFF AND VICE VERSA

CONTROL R - INITIALIZES "RECEIVE FILE" MODE

CONTROL T - INITIALIZES "TRANSMIT FILE" MODE

B. TRANSMIT FILE MODE

1. AUTOMATICALLY ISSUES ALL CP/CMS COMMANDS TO EFFECT THE TRANSFER OF AN ENTIRE FILE FROM FLOPPY DISK TO CP/CMS P-DISK

2. LINEFEED CHARACTERS APPEARING IN THE FLOPPY DISK FILES ARE FILTERED OUT; HOWEVER, TAB CHARACTERS ARE CONVERTED TO "?" AND TRANSMITTED TO CP/CMS

3. THE PRINTER DOES NOT WORK IN THIS MODE

NOTE: WHEN TRANSMITTING CONTINUOUS DATA FILES, THE PROGRAM SETS THE LINE LENGTH AT 132 CHARACTERS (83H). THE NAMED CMS FILETYPE MUST ACCOMMODATE THIS RECORD LENGTH. IF A SHORTER LINE LENGTH IS DESIRED, THE PROGRAM CAN BE ALTERED UNDER DDT AT PROGRAM COUNT OF 984H.

C. RECEIVE FILE MODE

1. AUTOMATICALLY ISSUES ALL CP/CMS COMMANDS TO EFFECT THE TRANSFER OF AN ENTIRE P-DISK FILE TO THE FLOPPY DISK

2. THE DATA BEING RECEIVED IS ECHOED TO THE CONSOLE FOR THE CONVENIENCE OF THE USER

3. THE TRANSMISSION BY CP/CMS CAN BE INTERRUPTED BY DEPRESSING ANY KEY. THIS ACTION RESTORES USER TO THE "DIRECT LINKUP" MODE AND THE CMS IS SHIFTED INTO CP. THE TERMINATED FILE IS LOST ALTHOUGH THE FILENAME WILL EXIST IN THE DISK DIRECTORY.

II. OPERATION

A. DIRECT LINKUP MODE

THE PROGRAM IS EXECUTED AS FOLLOWS:

LINK <CR>

THE USER IS PROMPTED WITH

DIAL 2721 FOR LINE -- TYPE CARRIAGE RETURN

A CONNECTED LINE IS INDICATED BY THE MESSAGE

CP-67 ON LINE

NORMAL LOGIN PROCEDURE AND CP/CMS TYPING CONVENTIONS ARE USED AND ANY KEY WILL "BREAK" THE CMS TRANSMISSIONS

B. TRANSMIT FILE MODE

UPON INITIALIZATION BY CONTROL T, USER IS PROMPTED WITH

DISK:FILENAME.FILETYPE

THE FILE TO BE TRANSMITTED SHOULD BE ENTERED EXACTLY ACCORDING TO THIS FORMAT. IF FORMAT IS VIOLATED, THE USER IS PROMPTED WITH

REPEAT

IF THE NAMED FILE CANNOT BE FOUND AS LISTED, THE APPROPRIATE PROMPT APPEARS

FILE NOT FOUND

AND USER IS RETURNED TO THE "DIRECT LINKUP" MODE. ASSUMING PROPER ENTRY OF THE FILE TO BE TRANSMITTED, THE NEXT PROMPT IS

CMS FILENAME FILETYPE?

THE FORMAT OF THE ANSWER TO THIS PROMPT IS NOT SPECIFIED BUT NOTE THAT THE CMS FILENAME WILL BE EXACTLY AS ENTERED.

NOTE: IF A KNOWN MISTAKE IS MADE IN ANSWERING THE ABOVE PROMPTS, TYPING CONTROL U WILL ALLOW USER TO START THE LINE AGAIN.

NOTE: THE CMS FILENAME SHOULD BE A NEW FILE SO THE CMS WILL SHIFT DIRECTLY INTO "INPUT" MODE.

AFTER ENTERING THE FILENAMES, THE PROGRAM OPERATES AUTOMATICALLY BUT ECHOES ITS COMMANDS TO CMS ON THE CONSOLE SO THE USER IS AWARE OF THE PROGRAM STATUS

NOTE: TYPING CONTROL D WILL IMMEDIATELY RETURN USER TO THE "DIRECT LINKUP" MODE

----- SAMPLE TRANSMITTED FILE -----

```
LINK:   DISK:FILENAME.FILETYPE
USER:   A:LINK.ASM<CR>
LINK:   CMS FILENAME FILETYPE?
USER:   HOOKER FORTRAN<CR>
LINK:   EDIT HOOKER FORTRAN
CMS:    >EDIT HOOKER FORTRAN
        >NEW FILE
        >INPUT:
LINK:   >TRANSMITTING
CMS:    >EDIT
LINK:   >SAVE
CMS:    >INPUT:
LINK:   >RELOADING
        TRANSMITTING
CMS:    >EDIT
LINK:   >FILE
CMS:    >R;
LINK:   >TRANSMISSION COMPLETE
        0034 RECORDS TRANSMITTED
        >
```

THE USER IS AUTOMATICALLY RETURNED TO THE "DIRECT LINKUP" MODE AT THIS TIME.

NOTE: IF THE FILE TO BE TRANSMITTED EXCEEDS THE BUFFER OF 40K BYTES, THE PROGRAM COMMANDS CMS TO SAVE THAT PORTION OF THE FILE, THEN 40K MORE BYTES ARE READ AND TRANSMITTED.

NOTE: FLOPPY DISK RECORDS ARE 128 BYTES IN LENGTH; P-DISK RECORDS ARE 829 BYTES IN LENGTH. DEPENDING ON THE CMS FILETYPE USED, ONE CMS RECORD EQUALS FROM ONE TO FOUR MDS RECORDS.

C. RECEIVE FILE MODE

UPON INITIALIZATION BY CONTROL R, THE FOLLOWING PROMPT APPEARS:

```
CMS FILENAME FILETYPE?
```

FORMAT REQUIREMENTS ARE SIMILAR TO THOSE ABOVE FOR "TRANSMIT FILE MODE". THE NEXT PROMPT IS

```
DISK:FILENAME.FILETYPE
```

AND AGAIN THE FORMAT IS THE SAME.

NOTE: THE FLOPPY DISK FILENAME AND FILETYPE SHOULD BE NEW TO THE DISK. THE PROGRAM WILL DELETE ANY EXISTING FILE WITH THE SPECIFIED FILENAME AND FILETYPE!!!!

IF DISK SPACE IS LIMITED, ONE OF THESE PROMPTS WILL APPEAR:

NO DIRECTORY SPACE AVAILABLE

(APPEARS BEFORE FILE IS TRANSMITTED BY CMS)

OR

DISK FULL

(APPEARS AFTER FILE HAS BEEN TRANSMITTED AND INDICATES FILE LENGTH EXCEEDED THE AVAILABLE DISK SPACE)

IN BOTH CASES, USER IS RETURNED TO THE "DIRECT LINKUP" MODE.

NOTE: TYPING CONTROL D WILL IMMEDIATELY RETURN USER TO THE "DIRECT LINKUP" MODE

ASSUMING NO DISK PROBLEMS, THE PROGRAM OPERATES AUTOMATICALLY.

----- SAMPLE RECEIVED FILE -----

```
LINK:   CMS FILENAME FILETYPE?
USER:   FOURPLAY OUTPUT72<CR>
LINK:   DISK:FILENAME.FILETYPE
USER:   HOWCUM.HEX<CR>
LINK:   PRINT FOURPLAY OUTPUT72
        RECEIVING
CMS:    :54424A2031303948534B37363231304D5F
        :ETC ETC ETC
        :ETC ETC
        :ETC
        >R;
LINK:   >TRANSMISSION COMPLETE
        0078 RECORDS TRANSMITTED
        >
```

THE USER IS AUTOMATICALLY RETURNED TO THE "DIRECT LINKUP" MODE.

NOTE: IF THE FILE TO BE RECEIVED FROM CMS EXCEEDS THE BUFFER SIZE OF 40K BYTES, THE REMAINDER OF THE FILE WILL BE LOST.

NOTE: IF USER ELECTS TO TERMINATE FILE RECEPTION, DEPRESSING ANY KEY WILL RETURN PROGRAM CONTROL TO "DIRECT LINKUP" AND THE CMS WILL BE INTERRUPTED

A HANDY REFERENCE GOUGE FOR "LINK" FOLLOWS:

LINK

<CR>	END OF LINE
RUBOUT	DELETE CHARACTER
CONTROL C	REBOOT
CONTROL D	RETURN TO DIRECT LINKUP
CONTROL I	TAB "?"
CONTROL P	PRINTER ON/OFF
CONTROL R	RECEIVE FILE MODE
CONTROL T	TRANSMIT FILE MODE
CONTROL U	DELETE LINE
BREAK	ANY KEY INTERRUPTS

MACK T. ELLIOTT, NPGS
AUGUST 22,1978

APPENDIX D

DATLINK USERS GUIDE

I. DATLINK IS A MODIFICATION OF THE LINK PROGRAM DESIGNED SPECIFICALLY FOR TRANSFERRING DATA FILES FROM FLOPPY DISK TO CP/CMS P-DISK.

A. DIRECT LINKUP MODE - THIS MODE OPERATES EXACTLY THE SAME AS IN THE LINK PROGRAM

B. TRANSMIT FILE MODE

1. DATA FILES ACQUIRED AND WRITTEN ON THE FLOPPY DISK BY THE GO PROGRAM ARE IN HEX CODE. THE FIRST FILE RECORD (128 BYTES) CONTAINS THE DATA FILENAME, INITIAL AND FINAL CHANNELS OF EACH SCAN, THE SCAN RATE, NUMBER OF DATA POINTS IN THE RUN, AND RUN COORDINATION NUMBER. ADDITIONALLY, THE FIRST FILE RECORD CONTAINS THE MOST SIGNIFICANT BYTE OF THE UPPER LIMIT ON MEMORY SPACE USED, AND THE NUMBER OF MEMORY BYTES USED PER SCAN (NUMBER OF CHANNELS TIMES TWO.

2. THE TRANSMIT FILE MODE ECHOES THE FILE PARAMETERS TO THE CONSOLE AND IMMEDIATELY BEGINS TRANSMISSION OF THE FILE TO CP/CMS. EACH HEX BYTE OF DATA IS CONVERTED TO TWO ASCII CHARACTERS BEFORE TRANSMISSION. THE LINE LENGTH IS SET AT THE NUMBER OF BYTES PER SCAN TO FACILITATE LATER FORMATTING FOR USE IN IBM 360 PROGRAMMING. E.G., THE MAXIMUM LINE LENGTH THAT COULD OCCUR WOULD BE 64 CHARACTERS (16 CHANNELS TIMES TWO BYTES PER CHANNEL TIMES TWO ASCII CHARACTERS PER BYTE).

3. THE MAXIMUM SIZED DATA FILE THAT CAN BE TRANSMITTED IS 40K (52K WITH A 62K SYSTEM) CORRESPONDING TO THE LARGEST DATA SAMPLE THAT CAN BE ACQUIRED WITH THE GO PROGRAM. ALSO, THE NUMBER OF FILE RECORDS TRANSMITTED IS NOT COUNTED AND DISPLAYED WITH THE DATLINK PROGRAM.

C. THE RECEIVE FILE MODE DOES NOT EXIST IN THE DATLINK PROGRAM.

II. OPERATION

A. DIRECT LINKUP MODE - EXECUTION OF THE DATLINK PROGRAM AND OPERATION OF THE "DIRECT LINKUP" MODE IS EXACTLY THE SAME AS FOR THE LINK PROGRAM.

B. TRANSMIT FILE MODE

UPON INITIALIZATION BY CONTROL T, THE PROMPTS AND REPLIES ARE THE SAME AS FOR THE LINK PROGRAM. BEFORE TRANSMISSION BEGINS, THE USER IS PROMPTED WITH THE DATA FILE PARAMETERS.

```
----- SAMPLE TRANSMITTED FILE -----

          DATLINK:  DISK:FILENAME.FILETYPE
          USER:     B:DATAØ3.XXX
          DATLINK:  CMS FILENAME FILETYPE
          USER:     FILE FTØ1FØØ1
          DATLINK:  DATAØ3
                    1Ø24 DATA POINTS
                    5ØØØ SCANS PER SECOND
                    RUN COORDINATION NUMBER 822ØØ1
                    EDIT FILE FTØ1FØØ1
          CMS:      >EDIT FILE FTØ1FØØ1
                    >NEW FILE
                    >DEFAULT PARAMETERS SET
                    >INPUT
          DATLINK:  >TRANSMITTING
          CMS:      >EDIT
          DATLINK:  >FILE
          CMS:      >R;
          DATLINK:  >TRANSMISSION COMPLETE
                    >
```

C. RECEIVE FILE MODE - UPON INITIALIZATION BY CONTROL R, THE USER IS PROMPTED WITH

TO RECEIVE FILE, USE LINK PROGRAM

THE MESSAGE IS SELF-EXPLANATORY

NOTE: ALL PROMPT REPLY FORMATS, ERROR MESSAGES, AND CONTROL CHARACTER USAGE IS EXACTLY THE SAME AS IN THE LINK PROGRAM.

M. T. ELLIOTT, NPGS
AUGUST 22, 1978

APPENDIX E

PRINT USER´S GUIDE

I. CAPABILITIES

A. PRINT INTERFACES THE INTEL MDS 800 WITH THE TELETYPE MODEL 40 HIGH SPEED PRINTER THROUGH AN INTEL SBC 534 INPUT/OUTPUT BOARD. PRINT ACCESSES FILES STORED ON FLOPPY DISK AND TRANSMITS THEM TO THE PRINTER AT A 9600 BAUD RATE.

B. FOR DISK FILES ALREADY FORMATTED, SUCH AS PRN FILES GENERATED BY THE TEX FORMATTER OR THE MACRO ASSEMBLER (PRODUCTS OF DIGITAL RESEARCH), THE PRINT PROGRAM OUTPUTS THE FILE WORD FOR WORD TO THE PRINTER.

C. ALL OTHER FILES STORED ON FLOPPY DISK IN ASCII CODE ARE FORMATTED BY PRINT FOR THE STANDARD 11 X 14 PAPER USED IN THE PRINTER. PRINT PROVIDES FOR ONE INCH MARGINS ON THE BOTTOM AND BOTH SIDES AND A THREE QUARTER INCH MARGIN AT THE TOP. EACH PAGE OF THE PRINTED FILE IS HEADED BY THE FILENAME, FILETYPE, AND PAGE NUMBER. PRINTED FILES ARE NORMALLY SINGLE SPACED, BUT A DOUBLE SPACE OPTION MAY BE SELECTED AND SHOULD COINCIDE WITH THE SPACING SWITCH ON THE PRINTER.

D. FOR PARTIAL PRINTOUTS OF LARGE FILES, THE USER CAN ENTER TWO STRINGS OF UP TO FIFTEEN CHARACTERS EACH, AND THE PROGRAM WILL SEARCH THE FILE AND PRINT ONLY THE TEXT BETWEEN THE STRINGS.

E. THE PRINT PROGRAM LOADS THE ENTIRE FILE INTO RANDOM MEMORY BEFORE COMMENCING OUTPUT TO THE PRINTER. IF THE AVAILABLE MEMORY (40K BYTES) IS EXCEEDED BY THE NAMED FILE, THEN 40K BYTES ARE PRINTED AND THEN ANOTHER 40K BYTES ARE LOADED AND PRINTED.

F. THE PRINT OPERATION CAN BE INTERRUPTED AT ANY TIME BY THE USER.

II. OPERATION

THE PRINT PROGRAM IS EXECUTED BY THE FOLLOWING COMMAND:

```
PRINT <DISK:>FILENAME.FILETYPE
```

THE PROGRAM TURNS ON THE PRINTER MOTOR AND SEARCHES FOR THE NAMED FILE. IF THE FILE CANNOT BE OPENED AS LISTED, THE FOLLOWING PROMPT APPEARS:

    FILE NOT FOUND

    DONE

AND THE USER MUST RE-EXECUTE USING THE CORRECT DISK/FILENAME/FILETYPE. AFTER THE FILE IS OPENED, USER IS PROMPTED WITH

    TEXT FILE?? (Y/N)

IF THE FILE HAS BEEN GENERATED BY THE TEX FORMATTER OR THE MACRO ASSEMBLER, NO FURTHER FORMATTING BY THE PRINT PROGRAM IS NEEDED. THE USER SHOULD TYPE YES (Y) AND THE FILE WILL PRINT AS FORMATTED. IF NO (N) IS SELECTED, THE NEXT PROMPT IS

    TYPE 2 FOR DOUBLE SPACE
    (DEFAULT = SINGLE SPACE)

TYPING ANY KEY OTHER THAN "2" WILL RESULT IN SINGLE SPACING.

NOTE: SELECTION OF DOUBLE SPACING MUST COINCIDE WITH THE SPACING SWITCH SETTING ON THE PRINTER.

NEXT THE USER IS PROMPTED WITH

    PRINT ALL (A) OR PART (P)??

IF ANY KEY OTHER THAN "P" IS SELECTED, THE PROGRAM WILL PRINT THE ENTIRE FILE. IF "P" IS SELECTED, ANOTHER PROMPT APPEARS:

    ENTER STRING1,STRING2

EITHER STRING MAY BE OMITTED, BUT THE COMMA MUST BE INCLUDED.

NOTE: THE PRINTOUT WILL INCLUDE THE FIRST STRING AND EXCLUDE THE SECOND STRING.

AT ANY TIME THE MODEL 40 IS PRINTING, USER MAY INTERRUPT BY TYPING ANY KEY. THE FOLLOWING PROMPT WILL APPEAR:

    TYPE K TO CANCEL OR SPACE TO CONTINUE

THIS MESSAGE IS SELF-EXPLANATORY.

AFTER COMPLETING THE PRINTOUT, THE PRINTER IS TURNED

OFF BY THE PROGRAM. THE FOLLOWING MESSAGE APPEARS ON THE CONSOLE:

DONE

A SOFT BOOT BY THE PROGRAM RESTORES USER TO CPM.

NOTE: IF THE PRINTER POWER SWITCH IS OFF OR THE PRINTER RUNS OUT OF PAPER, THE PRINT PROGRAM IDLES UNTIL THE CONDITION IS RECTIFIED, THEN RESUMES PRINTING.

M. T. ELLIOTT, NPGS
AUGUST 25, 1978

APPENDIX F

GO ASSEMBLY PROGRAM

```
;                         19 SEPT 1978     VERS 49
;**
;**MDS 8080 PROGRAM INTERFACES DATEL ST-800 ANALOG
;**TO DIGITAL CONVERTER BOARD AND INTEL DYNAMIC
;**MEMORY ACCESS CONTROLLER FOR HIGH SPEED DATA
;**ACQUISITION ---
;**MAXIMUM OF 16 CHANNELS ARE INPUT, CONVERTED,
;**AND STORED IN MEMORY AT A RATE OF 40 KHZ ---
;*********************************************
 0100           ORG 100H
 0100 C36003             JMP     START
;
;EQUATES
;
CR      EQU     0DH     ;CARRIAGE RETURN
LF      EQU     0AH     ;LINE FEED
BDOS    EQU     5H      ;BDOS ENTRY POINT
DMACMD  EQU     17H     ;DMA COMMAND WORD
REVRT   EQU     20H     ;CPU INTERRUPT CLEAR COMMAND
R04     EQU     20H     ;RESTART 04 ADDRESS
DMA     EQU     40H     ;DMA BASE ADDRESS
JUMP    EQU     0C3H    ;JUMP INSTRUCTION
MASK    EQU     0FCH    ;MASK ALTERATION PORT
MEMORY  EQU     0880H   ;DATA MEMORY BUFFER ADDRESS
;
;
;DATA SAVES
;
FLNAME: DB      0,'DATA01  XXX',0,0,0,0
        DS      17D
COUNT:  DS      1H      ;LENGTH REG VALUE
STACK:  DS      20H     ;SAVE ROOM FOR STACK
```

```
STKBTM  EQU     $       ;INITIATE STACK POINTER HERE
;
;
;MESSAGES
;
;
MSG1:   DB      CR,LF,LF,'ENTER STARTING CHANNEL $'
MSG2:   DB      CR,LF,'ENTER FINAL CHANNEL $'
MSG3:   DB      CR,LF,'START PULSE GENERATOR ---',CR,LF,LF,'$'
MSG4:   DB      CR,LF,'TRY AGAIN, TURKEY $'
M45:    DB      'DATA POINTS$'
MSG5:   DB      CR,LF,'ENTER DESIRED NUMBER OF DATA POINTS '
        DB      CR,LF,LF,'     ENTER   DATA POINTS     DISK SPACE'
        DB      CR,LF,' A           '
M5A:    DB      '1024             2K',CR,LF
        DB      '     B           '
M5B:    DB      '4096             8K',CR,LF
        DB      '     C          '
M5C:    DB      '10240           20K',CR,LF
        DB      '     D          '
M5D:    DB      '20480           40K',CR,LF
        DB      '     E          '
M5E:    DB      '26624           52K (62K SYSTEM)',CR,LF,'$'
MSG6:   DB      CR,LF,'ENTER '
M6A:    DB      'SCAN RATE $'
MSG65:  DB      CR,LF,'ENTER '
M65A:   DB      'COORDINATION NUMBER $'
MSG7:   DB      CR,LF,'WRITE DATA FILE ON DISK?? (Y/N) $'
MSG8:   DB      CR,LF,'ANOTHER DATA RUN DESIRED?? (Y/N) $'
MSG9:   DB      CR,LF,'DISK FULL - TRY ANOTHER - RETURN WHEN READY $'
MSG10:  DB      CR,LF,'DISK WRITE ERROR - TRY ANOTHER - RETURN WHEN READY$'
MSG11:  DB      CR,LF,'RUN COMPLETE - DISABLE PULSE',CR,LF,LF,'$'
;
;
;
;
```

```
                  START:
0360 314501               LXI     SP,     STKBTM  ;SET UP STACK POINTER
0363 3EC3                 MVI     A,      JUMP    ;JUMP INSTRUCTION
0365 322000               STA     R04             ;SET UP INTERRUPT
0368 218104               LXI     H,      RESET4  ;ADDR OF INT 4 ROUTINE
036B 222100               SHLD    R04+1
                  ;
                  ;CHANGE CPU MASK TO ACCEPT RST 04 INTERRUPTS
                  ;
036E 3E6E                 MVI     A,      6EH     ;ALLOWS RST 0,4,7
0370 D3FC                 OUT     MASK
                  ;
                  ;SPECIFY DISK DRIVE B FOR ALL DATA WRITES
                  ;
0372 0E0E                 MVI     C,      14
0374 1E01                 MVI     E,      1       ;DRIVE B
0376 CD0500               CALL    BDOS
                  ;
                  ;GET VALUES FOR INITIAL AND FINAL CHANNELS AND WORD LENGTH
                  ;
                  SETUP:
0379 CD4E05               CALL    RECORD          ;ZERO OUT RECORD LINE
037C CD9404               CALL    DIGIT1          ;GETS CHANNEL VALUES
037F 218508               LXI     H,      MEMORY+6;INITIAL CHANNEL VALUE
0382 3A8608               LDA     MEMORY+7        ;FINAL CHANNEL VALUE
0385 96                   SUB     M               ;DETERMINE DIFFERENCE
0386 F28F03               JP      DIFF
0389 CDFC04               CALL    OOPS            ;FINAL CAN'T BE LESS
038C C37903               JMP     SETUP           ;BACKUP AND TRY AGAIN
                  ;
                  DIFF:
038F C601                 ADI     1H              ;NUMBER WORDS PER
0391 17                   RAL                     ;SCAN IS NUMBER OF
0392 32B008               STA     MEMORY+30H      ;CHANS TIMES TWO
                  ;
                  DATPT:
```

```
                  ;
                  ;DETERMINE NUMBER OF DATA POINTS DESIRED
                  ;
Ø395 11B7Ø1               LXI     D,      MSG5     ;PROMPT USER
Ø398 ØEØ9                 MVI     C,      9H
Ø39A CDØ5ØØ               CALL    BDOS
Ø39D CDF4Ø4               CALL    KEY
                  ;
                  ;SEE WHICH CHOICE
                  ;
                  POINT:
03AØ FE41                 CPI     'A'              ;SEE IF A ENTERED
Ø3A2 CABFØ3               JZ      APOINT
Ø3A5 FE42                 CPI     'B'              ;SEE IF B ENTERED
Ø3A7 CAC7Ø3               JZ      BPOINT
Ø3AA FE43                 CPI     'C'              ;SEE IF C ENTERED
Ø3AC CACFØ3               JZ      CPOINT
Ø3AF FE44                 CPI     'D'              ;SEE IF D ENTERED
Ø3B1 CAD7Ø3               JZ      DPOINT
Ø3B4 FE45                 CPI     'E'              ;SEE IF E ENTERED
Ø3B6 CADFØ3               JZ      EPOINT
Ø3B9 CDFCØ4               CALL    OOPS             ;NOTHING ELSE IS VALID
Ø3BC C395Ø3               JMP     DATPT
                  ;
                  APOINT:
Ø3BF Ø1ØDØ2               LXI     B,      M5A
Ø3C2 3EØ8                 MVI     A,      9H
Ø3C4 C3E4Ø3               JMP     DOWN
                  BPOINT:
Ø3C7 Ø11CØ2               LXI     B,      M5B
Ø3CA 3E2Ø                 MVI     A,      21H
Ø3CC C3E4Ø3               JMP     DOWN
                  CPOINT:
Ø3CF Ø12AØ2               LXI     B,      M5C
Ø3D2 3E5Ø                 MVI     A,      51H
Ø3D4 C3E4Ø3               JMP     DOWN
```

```
                   DPOINT:
03D7 013902                 LXI     B,      M5D
03DA 3EA0                   MVI     A,      0A1H
03DC C3E403                 JMP     DOWN
                   EPOINT:
03DF 014802                 LXI     B,      M5E
03E2 3ED8                   MVI     A,      0D9H
                   DOWN:
03E4 C5                     PUSH    B
03E5 322401                 STA     COUNT           ;MSB OF WORD LENGTH
03E8 C609                   ADI     9H
03EA 32A008                 STA     MEMORY+20H      ;FOR THE RECORD
03ED 118708                 LXI     D,      MEMORY+7
03F0 CDF404                 CALL    KEY
03F3 C1                     POP     B
03F4 FE0D                   CPI     CR              ;WAIT FOR CARR RETURN
03F6 CAFF03                 JZ      DLOOP
03F9 CDFC04                 CALL    OOPS
03FC C39503                 JMP     DATPT
                   ;
                   DLOOP:
                   ;                COPY NUMBER DATA POINTS INTO RECORD
03FF 0A                     LDAX    B
0400 FE09                   CPI     09H             ;LOOK FOR TAB CHAR
0402 CA0B04                 JZ      DLEND
0405 12                     STAX    D
0406 03                     INX     B
0407 13                     INX     D
0408 C3FF03                 JMP     DLOOP
                   DLEND:
040B 3E24                   MVI     A,      '$'
040D 12                     STAX    D
040E 13                     INX     D
                   ;
                   ;GET PARAMETERS AND SAVE FOR THE RECORD
                   ;
```

```
                RATE:
040F D5                 PUSH   D
0410 116202             LXI    D,     MSG6
0413 0E09               MVI    C,     9H
0415 CD0500             CALL   BDOS
0418 D1                 POP    D
                RLOOP:
0419 CDF404             CALL   KEY              ;USER ENTERS SCAN
041C FE0D               CPI    CR               ;RATE TO BE USED-
041E CA2604             JZ     RLEND            ;THE ASCII CHARACTERS
0421 12                 STAX   D                ;ARE SAVED FOR THE
0422 13                 INX    D                ;RECORD
0423 C31904             JMP    RLOOP
                ;
                RLEND:
0426 3E24               MVI    A,     '$'       ;DELIMITER
0428 12                 STAX   D
0429 13                 INX    D
                ;
                ;GET RUN COORDINATION NUMBER FROM USER
                CNTRL:
042A D5                 PUSH   D
042B 117502             LXI    D,     MSG65
042E 0E09               MVI    C,     9H
0430 CD0500             CALL   BDOS
0433 D1                 POP    D
                ;
                CLOOP:
0434 CDF404             CALL   KEY              ;SAME AS ABOVE
0437 FE0D               CPI    CR
0439 CA4104             JZ     CLEND
043C 12                 STAX   D
043D 13                 INX    D
043E C33404             JMP    CLOOP
                ;
                CLEND:
```

```
0441 3E24              MVI     A,      '$'
0443 12                STAX    D
                 ;
                 BEGIN:
0444 CD5E04            CALL    DMASET          ;SETS UP DMA AND ST800
0447 D342              OUT     DMA+2H          ;RUNS ONE SCAN
                 ;
                 ;NOW READY TO BEGIN SCANNING WHEN PROMPTED
0449 117701            LXI     D,      MSG3
044C 0E09              MVI     C,      9H
044E CD0500            CALL    BDOS
                 ;
                 ;RESET DMA WORD LENGTH REG AND MEMORY ADDR REG,
                 ;CHANGE COMMAND WORD TO GIVE DMA COMPLETE
                 ;CONTROL OF THE SYSTEM BUS
                 ;
0451 D349              OUT     DMA+9H
0453 CD6E04            CALL    SYNC            ;RESETS DMA FOR RUN
0456 3E37              MVI     A,      DMACMD+00100000B
0458 D34A              OUT     DMA+0AH
                 ;
                 ;DATA ACQUISITION STARTS WITH PULSE GENERATOR
                 ;NOTHING TO DO BUT WAIT
                 ;
                 WAIT:
045A AF                XRA     A
045B C35A04            JMP WAIT
                 ;
                 ;END OF MAIN PROGRAM
                 ;
                 ;
                 ;
                 ;
                 ;
                 ;
                 ;
                 ;
```

```
                    ;SUBROUTINES
                    ;************************************************
                    ;
                    ;
                    ;*ROUTINE TO INITIALIZE AND RESET DMA AND ST-800
                    ; BOARDS -
                    ;*ST-800 IS ADDRESSED VIA DMA BOARD
                    ;*DMA IS SET UP TO GENERATE A LEVEL 4 INTERRUPT
                    ; WHENEVER ONE SCAN IS COMPLETED -
                    ;**********ADDRESS LISTING FOLLOWS**********
                    ;       DMA BASE ADDR             40H
                    ;       OUTPORT0/INPORT0          40H
                    ;       OUTPORT1/INPORT1          41H
                    ;       OUTPORT2                  42H
                    ;       DMA STATUS                46H
                    ;       DMA RESET                 49H
                    ;       DMA COMMAND               4AH
                    ;       LENGTH REGISTER (LSB)     4CH
                    ;       LENGTH REGISTER (MSB)     4DH
                    ;       MEMORY ADDR REG (LSB)     4EH
                    ;       MEMORY ADDR REG (MSB)     4FH
                    ;
                    ;
                    DMASET:
045E D349                   OUT     DMA+9H          ;RESET DMA
0460 3E17                   MVI     A,      DMACMD  ;ENABLES INTERRUPT,
0462 D34A                   OUT     DMA+0AH         ;8 BIT XFER TO MEMORY
0464 3A8508                 LDA     MEMORY+6H       ;START CHANNEL
0467 D340                   OUT     DMA
0469 3A8608                 LDA     MEMORY+7H       ;FINAL CHANNEL
046C D341                   OUT     DMA+1H
                    ;
                    ;ONE SCAN INITIATED BY "OUT 42" COMMAND -
                    ;THEN LENGTH REG AND MEMORY ADDR REG ARE RESET
                    ;BEFORE MAKING RUN - EACH SCAN IN THE RUN WILL
                    ;BE INITIATED BY THE PULSE GENERATOR OUTPUT
```

```
                  ;
                  SYNC:
046E AF                   XRA     A                 ;LSB OF LENGTH REG
046F D34C                 OUT     DMA+0CH
0471 3A2401               LDA     COUNT             ;MSB OF LENGTH REG
0474 D34D                 OUT     DMA+0DH
0476 210009               LXI     H,      MEMORY+80H
0479 7D                   MOV     A,      L         ;LSB OF MEMORY ADDR
047A D34E                 OUT     DMA+0EH
047C 7C                   MOV     A,      H         ;MSB OF MEMORY ADDR
047D D34F                 OUT     DMA+0FH
047F FB                   EI                        ;ENABLE INTERRUPTS
0480 C9                   RET
                  ;
                  ;DMA NOW READY TO GO WHEN COMMAND WORD IS ISSUED
                  ;
                  RESET4:
0481 D349                 OUT     DMA+9H            ;RESET DMA
0483 3E20                 MVI     A,      REVRT     ;CLEARS INT 4 FROM CPU
0485 D3FD                 OUT     0FDH              ;INTERRUPT PENDING STACK
0487 F1                   POP     PSW               ;KEEP STACK STRAIGHT
0488 113D03               LXI     D,      MSG11     ;GET USER TO TURN OFF
048B 0E09                 MVI     C,      9H        ;PULSE GENERATOR
048D CD0500               CALL    BDOS
0490 FB                   EI                        ;REENABLES INTERRUPTS
0491 C30705               JMP     DONE              ;GO PROCESS DATA
                  ;
                  ;ROUTINE TO READ IN INITIAL AND FINAL CHANNELS
                  ;
                  DIGIT1:
0494 114501               LXI     D,      MSG1
0497 0E09                 MVI     C,      9
0499 CD0500               CALL    BDOS              ;PROMPT USER
049C CDF404               CALL    KEY               ;GET ENTERED CHARACTER
049F FE0D                 CPI     CR
04A1 CA9404               JZ      DIGIT1
```

```
04A4 D630              SUI     30H             ;REDUCE ASCII
04A6 328508            STA     MEMORY+6
04A9 CDF404            CALL    KEY             ;SEE IF SECOND CHAR
04AC FE0D              CPI     CR
04AE CAC604            JZ      DIGIT2
04B1 D630              SUI     30H             ;REDUCE ASCII
04B3 C60A              ADI     0AH             ;CONVERT TO HEX
04B5 328508            STA     MEMORY+6
04B8 CDF404            CALL    KEY             ;STILL NEED CR
04BB FE0D              CPI     CR
04BD CAC604            JZ      DIGIT2
04C0 CDFC04            CALL    OOPS            ;TOO MANY CHARACTERS
04C3 C39404            JMP     DIGIT1          ;TRY AGAIN
                  ;
                  ;
                  DIGIT2:
04C6 116001            LXI     D,      MSG2
04C9 0E09              MVI     C,      9
04CB CD0500            CALL    BDOS            ;PROMPT USER
04CE CDF404            CALL    KEY             ;GET CHARACTER
04D1 FE0D              CPI     CR
04D3 CAC604            JZ      DIGIT2          ;CR NOT ALLOWED YET
04D6 D630              SUI     30H
04D8 328608            STA     MEMORY+7
04DB CDF404            CALL    KEY             ;GET NEXT CHAR
04DE FE0D              CPI     CR
04E0 C8                RZ                      ;FINISHED IF CR
04E1 D630              SUI     30H
04E3 C60A              ADI     0AH             ;CONVERT TO HEX
04E5 328608            STA     MEMORY+7
04E8 CDF404            CALL    KEY
04EB FE0D              CPI     CR
04ED C8                RZ                      ;FINISHED IF CR
04EE CDFC04            CALL    OOPS            ;TOO MANY CHARACTERS
04F1 C3C604            JMP     DIGIT2
                  ;
```

```
                 ;
                 ;
                 ;ROUTINE TO RETRIEVE CHARACTER FROM KEYBOARD
                 ;
                 KEY:
04F4 D5                  PUSH    D
04F5 0E01                MVI     C,      1H
04F7 CD0500              CALL    BDOS
04FA D1                  POP     D
04FB C9                  RET
                 ;
                 ;
                 ;ROUTINE PRINTS MESSAGE IF TOO MANY CHARACTERS
                 ;
                 OOPS:
04FC D5                  PUSH    D
04FD 119601              LXI     D,      MSG4
0500 0E09                MVI     C,      9
0502 CD0500              CALL    BDOS
0505 D1                  POP     D
0506 C9                  RET
                 ;
                 ;
                 DONE:
0507 119202              LXI     D,      MSG7    ;SEE IF USER WANTS
050A 0E09                MVI     C,      9H      ;FILE WRITTEN
050C CD0500              CALL    BDOS
050F CDF404              CALL    KEY             ;CHECK ANSWER
0512 FE4E                CPI     'N'             ;IF NO, CONTINUE
0514 CA2005              JZ      GETMOR
0517 CD0606              CALL    CRLF
051A CD0606              CALL    CRLF
051D C36C05              JMP     FLFILE          ;IF YES, GO WRITE
                 ;
                 ;
                 GETMOR:
```

```
0520 11B502             LXI     D,      MSG8        ;SEE IF USER WANTS
0523 0E09               MVI     C,      9H          ;ANOTHER RUN
0525 CD0500             CALL    BDOS
0528 CDF404             CALL    KEY                 ;CHECK ANSWER
052B FE59               CPI     'Y'                 ;IF YES, GO BACK
052D CA3305             JZ      RERUN
                ;
                ;OTHERWISE, ITS TIME TO QUIT
                ;
                EXIT:
0530 C30000             JMP     0H                  ;WARM BOOT
                ;
                ;SET UP FOR ANOTHER RUN
                ;
                RERUN:
0533 3A0901             LDA     FLNAME+6            ;INCREMENT FILE NAME
0536 3C                 INR     A
0537 320901             STA     FLNAME+6
053A FE3A               CPI     3AH
053C C27903             JNZ     SETUP
053F D60A               SUI     0AH
0541 320901             STA     FLNAME+6
0544 3A0801             LDA     FLNAME+5
0547 3C                 INR     A
0548 320801             STA     FLNAME+5
054B C37903             JMP     SETUP
                ;
                RECORD:
054E 3E00               MVI     A,      0H
0550 118008             LXI     D,      MEMORY
0553 0680               MVI     B,      80H
                RDLOOP:
0555 12                 STAX    D                   ;ZERO OUT FILE
0556 13                 INX     D                   ;RECORD WHICH WILL
0557 05                 DCR     B                   ;CONTAIN PROCESS
0558 C25505             JNZ     RDLOOP              ;INFORMATION
```

```
                  ;
                  ;FILL IN FILENAME
055B 010401               LXI     B,      FLNAME+1
055E 118008               LXI     D,      MEMORY
0561 2605                 MVI     H,      6H
                  RLOOP2:
0563 0A                   LDAX    B
0564 12                   STAX    D
0565 03                   INX     B
0566 13                   INX     D
0567 25                   DCR     H
0568 C26305               JNZ     RLOOP2
056B C9                   RET
                  ;
                  ;NEXT ROUTINE CREATES AND WRITES A DISK FILE -
                  ;THE FIRST FILE RECORD CONTAINS INFORMATION
                  ;WHICH WILL FACILITATE LATER RETRIEVAL OF THE
                  ;DATA ---
                  ;THE FIRST FILE RECORD CONTAINS THE DATA FILE
                  ;NAME, FIRST CHANNEL, FINAL CHANNEL, NUMBER
                  ;OF DATA POINTS, SCAN RATE, AND RUN CONTROL
                  ;NUMBER - ALSO THE NUMBER OF WORDS PER SCAN
                  ;THE REMAINDER OF THE FIRST FILE RECORD IS ZEROES
                  ;
                  ;
                  FLFILE:
                  ;
                  ;CREATE FILE ON DISK DRIVE B
                  ;
056C 0E13                 MVI     C,      19
056E 110301               LXI     D,      FLNAME
0571 CD0500               CALL    BDOS            ;DELETE OLD FILE, SAME NAME
                  ;CLEAN UP FILE CONTROL BLOCK
0574 AF                   XRA     A
0575 320F01               STA     FLNAME+12
0578 321001               STA     FLNAME+13
```

```
057B 321101         STA     FLNAME+14
057E 321201         STA     FLNAME+15
               ;
               ;CREATE NEW FILE
0581 0E16           MVI     C,      22
0583 110301         LXI     D,      FLNAME
0586 CD0500         CALL    BDOS            ;CREATE NEW FILE
0589 FEFF           CPI     255             ;RETURNS 255 IF NOT
058B CA2706         JZ      NOROOM          ;ENOUGH DISK SPACE
058E AF             XRA     A               ;ZERO IT
058F 322301         STA     FLNAME+32       ;NEXT RECORD COUNT
               ;
               ;WHILE DISK WRITE OCCURS, ECHO DATA FILE PARAMETERS
               ;TO CONSOLE FOR CORRELATION
               ;
0592 11AB01         LXI     D,      M45
0595 0E09           MVI     C,      9H
0597 CD0500         CALL    BDOS
059A CD0606         CALL    CRLF
059D 118708         LXI     D,      MEMORY+7
05A0 CD1706         CALL    CONSL
05A3 CD0606         CALL    CRLF
05A6 D5             PUSH    D
05A7 116A02         LXI     D,      M6A
05AA 0E09           MVI     C,      9H
05AC CD0500         CALL    BDOS
05AF D1             POP     D
05B0 CD0606         CALL    CRLF
05B3 CD1706         CALL    CONSL
05B6 CD0606         CALL    CRLF
05B9 D5             PUSH    D
05BA 117D02         LXI     D,      M65A
05BD 0E09           MVI     C,      9H
05BF CD0500         CALL    BDOS
05C2 D1             POP     D
05C3 CD0606         CALL    CRLF
```

```
Ø5C6 CD17Ø6              CALL    CONSL
Ø5C9 CDØ6Ø6              CALL    CRLF
                ;
                ;
                ;
                ;SINCE DMA PUT PAIRS OF DATA BYTES INTO MEMORY IN REVERSE
                ;ORDER, WANT TO REVERSE THEM BEFORE WRITING ON DISK
                ;
                FLIP:
Ø5CC 3AAØØ8              LDA     MEMORY+2ØH                  ;UPPER LIMIT ON MEMORY
Ø5CF 21ØØØ9              LXI     H,      MEMORY+8ØH          ;BEGINNING OF DATA
                ;
                FLOP:
Ø5D2 46                  MOV     B,      M       ;GET LSB
Ø5D3 23                  INX     H
Ø5D4 4E                  MOV     C,      M       ;GET MSB
Ø5D5 7Ø                  MOV     M,      B       ;PUT LSB
Ø5D6 2B                  DCX     H
Ø5D7 71                  MOV     M,      C       ;PUT MSB
Ø5D8 23                  INX     H
Ø5D9 23                  INX     H
Ø5DA BC                  CMP     H               ;CHECK AGAINST LIMIT
Ø5DB C2D2Ø5              JNZ     FLOP
                ;
                ;DATA PAIRS NOW IN CORRECT ORDER
                ;
                ;
                ;READY TO START WRITING ONTO DISK
                ;
                FWRITE:
Ø5DE 118ØØ8              LXI     D,      MEMORY  ;INFO RECORD
                ;
                FLOOP:
Ø5E1 D5                  PUSH    D               ;SAVE POINTER
Ø5E2 ØE1A                MVI     C,      26      ;
```

```
05E4 CD0500          CALL    BDOS                   ;CHANGE BUFFER ADDRESS
05E7 110301          LXI     D,         FLNAME
05EA 0E15            MVI     C,         21
05EC CD0500          CALL    BDOS                   ;WRITE ONE RECORD
05EF D1              POP     D                      ;RETRIEVE POINTER
05F0 F5              PUSH    PSW                    ;WILL CHECK LATER
05F1 218000          LXI     H,         80H
05F4 19              DAD     D                      ;INCREMENT POINTER
05F5 EB              XCHG                           ;BY 80H
05F6 F1              POP     PSW
05F7 FE00            CPI     0H                     ;CHECK FOR WRITE ERRORS
05F9 C23506          JNZ     ERROR
05FC 3AA008          LDA     MEMORY+20H             ;CHECK END OF DATA
05FF BA              CMP     D                      ;MSB ONLY
0600 CA4806          JZ      CLOSE
0603 C3E105          JMP     FLOOP                  ;GO DO ANOTHER RECORD
                ;
                ;THIS CONTINUES UNTIL ALL DATA WRITTEN ONTO DISK
                ;
                ;ROUTINE PUTS CARRIAGE RETURN, LINE FEED ON CONSOLE
                CRLF:
0606 D5              PUSH    D
0607 1E0D            MVI     E,         CR
0609 0E02            MVI     C,         2H
060B CD0500          CALL    BDOS
060E 1E0A            MVI     E,         LF
0610 0E02            MVI     C,         2H
0612 CD0500          CALL    BDOS
0615 D1              POP     D
0616 C9              RET
                ;
                ;ROUTINE PRINTS DATA STRINGS ON CONSOLE
                CONSL:
0617 1A              LDAX    D
0618 13              INX     D
0619 FE24            CPI     '$'
```

```
Ø61B C8                RZ
Ø61C D5                PUSH    D
Ø61D 5F                MOV     E,      A
Ø61E ØEØ2              MVI     C,      2H
Ø62Ø CDØ5ØØ            CALL    BDOS
Ø623 D1                POP     D
Ø624 C317Ø6            JMP     CONSL
                ;
                ;ROUTINE INFORMS USER THAT DISK OR DIRECTORY IS FULL
                ;
                ;
                NOROOM:
Ø627 11D9Ø2            LXI     D,      MSG9
Ø62A ØEØ9              MVI     C,      9H
Ø62C CDØ5ØØ            CALL    BDOS
Ø62F CDF4Ø4            CALL    KEY             ;WAIT FOR RESPONSE
Ø632 C36CØ5            JMP     FLFILE          ;TRY ANOTHER WRITE
                ;
                ;
                ;
                ERROR:
Ø635 FEØ2              CPI     2               ;SEE IF DISK FULL
Ø637 CA27Ø6            JZ      NOROOM
Ø63A 11Ø8Ø3            LXI     D,      MSG1Ø
Ø63D ØEØ9              MVI     C,      9H
Ø63F CDØ5ØØ            CALL    BDOS            ;INFO USER OF ERROR
Ø642 CDF4Ø4            CALL    KEY             ;CHECK FOR RESPONSE
Ø645 C36CØ5            JMP     FLFILE          ;
                ;
                ;IF ERROR OCCURRED IN WRITING ON DISK, ANOTHER WRITE SHOULD
                ;BE ATTEMPTED ON ANOTHER DISK
                ;
                ;
                ;WHENEVER DATA WRITE IS COMPLETED, NEED TO CLOSE FILE
                ;
                CLOSE:
```

```
Ø648 11Ø3Ø1          LXI    D,     FLNAME
Ø64B ØE1Ø            MVI    C,     16
Ø64D CDØ5ØØ          CALL   BDOS
Ø65Ø C32ØØ5          JMP    GETMOR        ;CHECK WITH USER
                ;
                ;
                ;
                ;
                ;
                ;
                ;**********************************************
Ø653            END 1ØØH
```

APPENDIX G

LINK ASSEMBLY PROGRAM

```
                      ;       UPDATED 1200 ON 26 APR 78
 0100                         ORG     100H
 0100 C30D04                  JMP     START
BDOS     EQU      5H          ;ENTRY POINT
XON      EQU      11H         ; END OF LINE FROM VIRTUAL MACHINE
XOFF     EQU      13H         ;END OF LINE TO VIRTUAL MACHINE
CR       EQU      0DH         ;CARRIAGE RETURN
LF       EQU      0AH         ;LINE FEED
FF       EQU      0CH         ;FORM FEED
EOF      EQU      1AH         ;END OF FILE CHAR FOR DISK WRITE
RUB      EQU      7FH         ;DELETE CHARACTER
CNTLC    EQU      03H         ;WARM BOOT
CNTLD    EQU      04H         ;RESTORES "DIRECT LINKUP" MODE
CNTLG    EQU      07H         ;PRINT INSTRUCTIONS
CNTLI    EQU      09H         ;TAB CHARACTER
CNTLP    EQU      10H         ;CONTROL P TURNS PRINTER ON AND OFF
CNTLR    EQU      12H         ;CONTROL R FOR RECEIVE FILE
CNTLT    EQU      14H         ;CONTROL T FOR TRANSMIT FILE
CNTLU    EQU      15H         ;DELETE LINE
FLIMIT   EQU      230H        ;ALLOWS 304 RECORDS OF 128 BYTES
BUFFMAX  EQU      0D000H      ;MAX SIZE OF TRANSFERRED FILE
COUNT:   DS       2           ;COUNT OF RECORDS TRANSFERRED
FCOUNT:  DS       2           ;FILE COUNT RECORD
PPREG    DS       1           ;PRINTER CONTROL REG;0 OFF,1 ON
```

```
MSG1:    DB      CR,LF,'DIAL 2721 FOR LINE--CONTROL G FOR INSTRUCTIONS',CR,LF,'$'
MSG2:    DB      '                       CONTROL C - REBOOT',CR,LF
         DB      '                       CONTROL D - RETURN TO DIRECT LINKUP',CR,LF
         DB      '                       CONTROL G - INSTRUCTIONS',CR,LF
         DB      '                       CONTROL I - TAB',CR,LF
         DB      '                       CONTROL P - PRINTER ON/OFF',CR,LF
         DB      '                       CONTROL R - RECEIVE FILE',CR,LF
         DB      '                       CONTROL T - TRANSMIT FILE',CR,LF
         DB      '                       CONTROL U - DELETE LINE',CR,LF
         DB      '                       RUBOUT    - DELETE CHARACTER',CR,LF
         DB      '                       XMIT      - INTERRUPT CMS',CR,LF,'$'
MSG3:    DB      CR,LF,'DISK:FILENAME.FILETYPE',CR,LF,'$'
MSG4:    DB      CR,LF,'REPEAT',CR,LF,'$'
MSG5:    DB      'EDIT $'
MSG5A:   DB      'FILE NOT FOUND',CR,LF,'>$'
MSG6:    DB      'TRANSMITTING',CR,LF,'$'
MSG7:    DB      'TRANSMISSION COMPLETE',CR,LF,'$'
MSG8:    DB      'FILE$'
MSG1Ø:   DB      'PRINT $'
MSG11:   DB      'NO DIRECTORY SPACE AVAILABLE',CR,LF,'>$'
MSG12:   DB      'RECEIVING',CR,LF,'$'
MSG13:   DB      'DISK FULL',CR,LF,'$'
MSG14:   DB      ' RECORDS TRANSFERRED',CR,LF,'>$'
MSG15:   DB      'CMS FILENAME FILETYPE?',CR,LF,'$'
MSG17:   DB      'FILE EXCEEDS BUFFER - ONLY 52K BYTES TRANSFERRED',CR,LF,'$'
MSG18:   DB      'RELOADING',CR,LF,'$'
MSG19:   DB      'SAVE$'
STACK:   DS      2Ø
STKBTM   EQU     $

Ø4ØD 31ØDØ4      START:  LXI     SP,     STKBTM
Ø41Ø 3EØØ                MVI     A,      Ø
Ø412 32Ø7Ø1              STA     PPREG           ;INITIALLY PRINTER IS OFF
Ø415 11Ø8Ø1              LXI     D,      MSG1
Ø418 CDACØ7              CALL    MESSAGE         ;PROMPTS USER TO CALL FOR LINE
```

```
041B CDAE05              CALL    BOARD          ;INITIALIZES SBC 534 BOARD
041E DB60                IN      60H
                         ;
                         ;TRANSMIT MODE
                         ;
                TX:
0420 DB61                IN      61H            ;CHECKS LINE FOR MESSAGE
0422 E602                ANI     2
0424 C2A904              JNZ     CRCV1
0427 DBF7                IN      0F7H           ;CHECKS KEYBOARD
0429 E602                ANI     2
042B CA2004              JZ      TX             ;LOOPS UNTIL ONE OF THE ABOVE
042E 0E01                MVI     C,      1
0430 CD0500              CALL    BDOS           ;READ CHAR FROM CONSOLE
0433 FE0D                CPI     CR             ;CHECK FOR CR
0435 CA9204              JZ      RCV            ;SWITCH TO RECEIVE MODE
0438 FE10                CPI     CNTLP
043A CA0305              JZ      PRTCONT        ;TURN PRINTER ON/OFF
043D FE12                CPI     CNTLR
043F CA3D06              JZ      FILERX         ;RECEIVE FILE MODE
0442 FE14                CPI     CNTLT
0444 CA0B06              JZ      FILETX         ;TRANSMIT FILE MODE
0447 FE03                CPI     CNTLC
0449 CA0000              JZ      00H            ;ESCAPE BY REBOOTING
044C FE07                CPI     CNTLG          ;PRINT INSTRUCTIONS
044E CAE604              JZ      GOUGE
0451 FE09                CPI     CNTLI
0453 CC8C04              CZ      CHNG4          ;TRANSMIT TAB CHAR "?"
0456 FE7F                CPI     RUB
0458 CC7C04              CZ      CHNG2          ;TRANSMIT DELETE CHAR SYMBOI "@"
045B FE15                CPI     CNTLU
045D CA8404              JZ      CHNG3          ;TRANSMIT DELETE LINE SYMBOL "["
                                                ;AND XOFF
0460 4F                  MOV     C,      A
0461 FE11                CPI     XON
0463 CA7204              JZ      CTX
```

```
Ø466 3AØ7Ø1              LDA     PPREG              ;CHECK IF PRINTER ON
Ø469 FEØØ                CPI     Ø
Ø46B CA72Ø4              JZ      CTX
Ø46E 79                  MOV     A,      C
Ø46F CD2CØ5              CALL    DRIVER
                 CTX:
Ø472 79                  MOV     A,      C
Ø473 CD6FØ5              CALL    SEND               ;SENDS CHAR TO VIRTUAL MACHINE
Ø476 C32ØØ4              JMP     TX                 ;LOOPS FOREVER
                 CHNG1:
Ø479 3E3F                MVI     A,      '?'
Ø47B C9                  RET
                 CHNG2:
Ø47C 3EØ8                MVI     A,      Ø8H        ;BACKSPACE
Ø47E CD37Ø5              CALL    CONOUT
Ø481 3E4Ø                MVI     A,      '@'
Ø483 C9                  RET
                 CHNG3:
Ø484 3E5B                MVI     A,      '['
Ø486 CD6FØ5              CALL    SEND
Ø489 C392Ø4              JMP     RCV
                 CHNG4:
Ø48C 3E3F                MVI     A,      '?'
Ø48E CD37Ø5              CALL    CONOUT
Ø491 C9                  RET
                         ;
                         ;RECEIVE MODE
                         ;
                 RCV:
Ø492 3AØ7Ø1              LDA     PPREG              ;CHECK IF PRINTER ON
Ø495 FEØØ                CPI     Ø
Ø497 CAA4Ø4              JZ      CRCV
Ø49A 3EØD                MVI     A,      CR         ;START NEW LINE ON PRINTER
Ø49C CD2CØ5              CALL    DRIVER
Ø49F 3EØA                MVI     A,      LF
Ø4A1 CD2CØ5              CALL    DRIVER
```

```
                CRCV:
04A4 3E13               MVI     A,      XOFF    ;END OF LINE CHAR
04A6 CD6F05             CALL    SEND
                CRCV1:
                ;HL REGISTER POINTS TO ADDR FOR NEXT WORD RECEIVED
                ;DE REGISTER POINTS TO ADDR OF NEXT WORD TO BE PRINTED
04A9 215A0A             LXI     H,      BUFF    ;FIFO BUFFER ADDR
04AC 115A0A             LXI     D,      BUFF
                RX1:
04AF CD7A05             CALL    BREAK
04B2 DB61               IN      61H             ;CHECK LINE FOR CHAR
04B4 E602               ANI     02H
04B6 CA4205             JZ      CKPRT           ;IF LINE NOT READY, CHECK IF
                                                ;BUFFER CAUGHT UP

                RX:
04B9 DB60               IN      60H             ;INPUT WORD FROM LINE
04BB E67F               ANI     7FH
04BD FE11               CPI     XON
04BF CACC04             JZ      CATCH           ;END OF LINE - LET BUFFER
                                               ;CATCH UP
04C2 FE13               CPI     XOFF
04C4 CAAF04             JZ      RX1             ;FILTER OUT XOFF CHAR
04C7 77                 MOV     M,      A       ;STORE CHAR
04C8 23                 INX     H
04C9 C3AF04             JMP     RX1             ;LOOP UNTIL END OF LINE
                CATCH:
04CC 77                 MOV     M,      A       ;STORE LAST WORD

                LOOOP:
04CD 1A                 LDAX    D               ;NEXT WORD TO BE PRINTED
04CE FE11               CPI     XON
04D0 CA2004             JZ      TX              ;GO BACK TO TRANSMIT MODE
04D3 CD3705             CALL    CONOUT          ;PRINT ON CONSOLE
04D6 3A0701             LDA     PPREG           ;CHECK IF PRINTER ON
04D9 FE00               CPI     0
```

```
04DB CAE204            JZ      BACK
04DE 1A                LDAX    D
04DF CD2C05            CALL    DRIVER
              BACK:
04E2 13                INX     D
04E3 C3CD04            JMP     LOOOP                ;LOOP UNTIL CAUGHT UP
              GOUGE:
04E6 113D01            LXI     D,       MSG2
              GLOOP:
04E9 1A                LDAX    D
04EA FE24              CPI     '$'
04EC CA2004            JZ      TX
04EF CD3705            CALL    CONOUT
04F2 47                MOV     B,       A
04F3 3A0701            LDA     PPREG
04F6 FE00              CPI     0
04F8 CAFF04            JZ      GLP
04FB 78                MOV     A,       B
04FC CD2C05            CALL    DRIVER
              GLP:
04FF 13                INX     D
0500 C3E904            JMP     GLOOP
              PRTCONT:
0503 3A0701            LDA     PPREG                ;CHECK IF PRINTER ON OR OFF
0506 FE00              CPI     0
0508 C22005            JNZ     PRTOFF               ;IF ON, WANT TO TURN OFF
050B CDE505            CALL    USART2
050E 3E01              MVI     A,       1
0510 320701            STA     PPREG                ;LATER ROUTINES CHECK THIS ADDR
0513 3E0D              MVI     A,       CR          ;START PRINTER ON NEW LINE
0515 CD2C05            CALL    DRIVER
0518 3E0A              MVI     A,       LF
051A CD2C05            CALL    DRIVER
051D C32004            JMP     TX                   ;RETURN TO TRANSMIT MODE
              PRTOFF:
0520 3E30              MVI     A,       30H         ;CONTROL WORD - TURN PRINTER OFF
```

```
0522 D363             OUT     63H
0524 3E00             MVI     A,      0
0526 320701           STA     PPREG               ;LATER ROUTINES CHECK THIS ADDR
0529 C32004           JMP     TX
           ;ROUTINE TO DRIVE PRINTER USART
           DRIVER:
052C F5               PUSH    PSW
           SLO:
                     IN      63H                 ;WAIT UNTIL XMITTER READY
052F 0F               RRC
0530 D22D05           JNC     SLO
0533 F1               POP     PSW
0534 D362             OUT     62H
0536 C9               RET

           ;ROUTINE TO DRIVE CONSOLE USART
           CONOUT:
0537 F5               PUSH    PSW
            SLO2:
0538 DBF7             IN      0F7H
053A 0F               RRC
053B D23805           JNC     SLO2
053E F1               POP     PSW
053F D3F6             OUT     0F6H
0541 C9               RET
            ;KEEPS TRACK OF WHICH RECEIVED DATA HAS BEEN PRINTED
           CKPRT:
0542 7D               MOV     A,      L
0543 BB               CMP     E
0544 CAAF04           JZ      RX1                 ;CAUGHT UP, NO NEED TO PROCEED
0547 DBF7             IN      0F7H
0549 0F               RRC
054A D2AF04           JNC     RX1                 ;CONSOLE NOT READY - NO NEED
                                                 ;TO PROCEED
054D 3A0701           LDA     PPREG               ;CHECK IF PRINTER ON
0550 FE00             CPI     0
```

```
0552 CA5B05          JZ      CKP2              ;IF PRINTER NOT ON, NO NEED
                                               ;TO PROCEED
0555 DB63            IN      63H
0557 0F              RRC
0558 D2AF04          JNC     RX1               ;IF PRINTER NOT READY, NO NEED
                                               ;TO PROCEED
            CKP2:
055B 1A              LDAX    D                 ;NEXT WORD TO BE PRINTED
055C D3F6            OUT     0F6H              ;OUT TO CONSOLE
055E D362            OUT     62H               ;OUT TO PRINTER
0560 13              INX     D
0561 7D              MOV     A,     L          ;CHECK AGAIN TO SEE IF BUFFER IS
0562 BB              CMP     E                 ;CAUGHT UP - IF SO, RESET BUFFER
0563 C2AF04          JNZ     RX1
0566 215A0A          LXI     H,     BUFF
0569 115A0A          LXI     D,     BUFF
056C C3AF04          JMP     RX1
            ;DRIVES USART ON HIGH SPEED LINE
            SEND:
056F F5              PUSH    PSW
            WAIT:
0570 DB61            IN      61H
0572 0F              RRC
0573 D27005          JNC     WAIT
0576 F1              POP     PSW
0577 D360            OUT     60H
0579 C9              RET
            ;CHECKS KEYBOARD FOR INTERRUPT
            BREAK:
057A DBF7            IN      0F7H
057C E602            ANI     2
057E C8              RZ                        ;IF NONE, GO BACK TO RECEIVE
057F DBF6            IN      0F6H              ;INTERPT PRESENT-CHECK FOR BREAK
0581 E67F            ANI     7FH
0583 FE11            CPI     XON
0585 C0              RNZ                       ;IGNORE IF NOT BREAK
```

```
0586 3E3F               MVI     A,      3FH     ;CONTROL - DRIVES XMIT LINE LOW
0588 D361               OUT     61H             ;HOLD LINE LOW FOR 2 WORDLENGTHS
058A 010004             LXI     B,      400H    ;WAIT 10 MILLISECS
                DLA1:
058D 0B                 DCX     B
058E 78                 MOV     A,      B
058F FE00               CPI     0
0591 CA9F05             JZ      DLA3
0594 DB61               IN      61H             ;CHECK LINE FOR CHAR
0596 E602               ANI     2
0598 CA8D05             JZ      DLA1
059B DB60               IN      60H
059D 77                 MOV     M,      A
059E 23                 INX     H
                DLA3:
059F 01A005             LXI     B,      5A0H    ;DELAY 16 MILLISEC
                DLA2:
05A2 0B                 DCX     B
05A3 78                 MOV     A,      B
05A4 FE00               CPI     0
05A6 C2A205             JNZ     DLA2
                RESET:
05A9 3E37               MVI     A,      37H
05AB D361               OUT     61H
05AD C9                 RET

                BOARD:

;
;   THIS ROUTINE INITIALIZES THE 534 BOARD, THE TIMERS, AND THE TWO USARTS
;   NEEDED TO DRIVE THE IBM HIGH SPEED LINE AND THE MODEL 40 PRINTER
;
;   BASE ADDR OF 534 BOARD          60H
;   CMD ADDR OF LINE USART          61H
;   DATA ADDR OF LINE USART         60H
;   CMD ADDR OF PTR USART           63H
```

```
;   DATA ADDR OF PTR USART         62H
;
;   TWO MORE USARTS AND ONE 8255 PARALLEL INTERFACE AND THEIR TIMERS ARE
;   AVAILABLE ON THE 534 BOARD.  NEW INTERFACES MUST BE PROGRAMMED BEFORE USE
;
;
Ø5AE F3               DI                              ;DISABLES 8080 INTERRUPTS
Ø5AF D36F             OUT     6FH                     ;RESETS BOARD
Ø5B1 D36C             OUT     6CH                     ;SELECTS BOARD CONTROL BLOCK
Ø5B3 CDBBØ5           CALL    TIMER                   ;INITIALIZE PIT CHIPS
Ø5B6 CDD8Ø5           CALL    USART                   ;INITIALIZE USARTS
Ø5B9 FB               EI                              ;REENABLES INTERRUPTS
Ø5BA C9               RET

              ;
              ;
              ;MUST SET UP TIMER CHIPS ACCORDING TO PAGE 3-12 OF 534 MANUAL
              ;  CHIP Ø HAS THREE TIMERS ON IT
              ;  TIMERS Ø AND 1 OF CHIP Ø ARE CONNECTED TO USARTS 1 AND 2
              ;  RESPECTIVELY, DRIVING THE IBM LINE AND THE PRINTER
              ;
              TIMER:
Ø5BB D36C             OUT     6CH                     ;SELECT BOARD CONTROL BLOCK
Ø5BD 3E36             MVI     A,      36H             ;SELECT TIMER Ø FOR LINE USART
Ø5BF D363             OUT     63H                     ;ADDR OF COUNTER Ø MODE CONTROL
Ø5C1 3E4Ø             MVI     A,      4ØH             ;
Ø5C3 D36Ø             OUT     6ØH                     ;SET N-4ØH IN TIMER Ø
Ø5C5 3EØØ             MVI     A,      ØH              ;CCLK/N=19.2KHZ FOR 12ØØ BAUD,
Ø5C7 D36Ø             OUT     6ØH                     ;BRF=16X
Ø5C9 3E76             MVI     A,      76H             ;
Ø5CB D363             OUT     63H                     ;SELECT TIMER 1 FOR PTR USART
Ø5CD 3EØ8             MVI     A,      8H              ;
Ø5CF D361             OUT     61H                     ;SET N=8 IN TIMER 1
Ø5D1 3EØØ             MVI     A,      ØH              ;CCLK/N=153.6KHZ FOR 96ØØ BAUD,
Ø5D3 D361             OUT     61H                     ;BRF=16X
Ø5D5 D36D             OUT     6DH                     ;PUTS BOARD IN DATA BLOCK
```

```
05D7 C9                 RET

                  ;
                  ;
                  ;SET UP BOTH USARTS WITH RESETS AND MODE WORDS
                  ;
                  USART:
04D9 FE00               CPI     0
05D8 3ECA               MVI     A,      0CAH    ;2 STOP, PAR DISABLED, 7 BITS
05DA D361               OUT     61H
05DC 3E5A               MVI     A,      5AH     ;1 STOP, PAR DISABLED, 7 BITS
05DE D363               OUT     63H
05E0 3E37               MVI     A,      37H
05E2 D361               OUT     61H
05E4 C9                 RET
                  USART2:
05E5 3E33               MVI     A,      33H
05E7 D363               OUT     63H
05E9 C9                 RET
             ;THIS SECTION PERTAINS TO TRANSFERRING COMPLETE
             ;FILES BETWEEN MDS AND IBM 360
             ;
FCB     EQU     5CH             ;FCB ADDR
FCBCN   EQU     FCB+0           ;DISK NAME
FCBFN   EQU     FCB+1           ;FILENAME(8CHAR)
FCBFT   EQU     FCB+9           ;FILETYPE (3CHAR)
FCBRL   EQU     FCB+12          ;REEL NUMBER
FCBRC   EQU     FCB+15          ;FILE RECORD COUNT (0-127)
FCB2:   DS      33              ;NEW FILENAME AND FILETYPE
FCBCR   EQU     FCB+32          ;NEXT RECORD NUMBER

                  ;SUBR PROMPTS CONSLE FOR FILE TO BE XMITTED, SETS UP FILE
                  ;CONTROL BLOCK, OPENS NEW CMS FILE, TRANSMITS FILE, AND
                  ;RETURNS USER TO DIRECT CMS LINKUP
                  ;
```

```
                    FILETX:
Ø60B 3EØØ                   MVI     A,      Ø
Ø60D 32Ø3Ø1                 STA     COUNT
Ø61Ø 32Ø4Ø1                 STA     COUNT+1
Ø613 CD69Ø6                 CALL    RESTRT          ;SETS UP FILE CONTROL BLOCK
Ø616 CD7DØ7                 CALL    CRLF
Ø619 CD1CØ7                 CALL    CPNAME          ;CP/CMS FILENAME, FILETYPE
Ø61C CD7DØ7                 CALL    CRLF
Ø61F CD8CØ7                 CALL    OPEN            ;OPENS DISK FILE
Ø622 CDB2Ø7                 CALL    FILERD          ;READS DISK FILE
Ø625 CDØFØ9                 CALL    CMS             ;PREPARES CMS TO RECEIVE FILE
Ø628 CD3BØ9                 CALL    ANS             ;WAITS FOF ANSWER
Ø62B CD78Ø9                 CALL    XMIT            ;TRANSMITS FILE
Ø62E CD3BØ9                 CALL    ANS
Ø631 CD28ØA                 CALL    FILE            ;"FILES" FILE IN CMS
Ø634 CD3BØ9                 CALL    ANS
Ø637 CD46Ø8                 CALL    TALLY           ;PRINTS OUT RECORD COUNT
Ø63A C32ØØ4                 JMP     TX              ;RETURNS TO TRANSMIT MODE


                     ;SUBR PROMPTS CONSOLE FOR FILE TO BE RECEIVED, SETS UP FILE
                     ;CONTROL BLOCK AND CREATES FILE ON FLOPPY DISK, RECEIVES FILE
                     ;FROM CMS AND ECHOES ON CONSOLE, CLOSES FILE AND RESTORES
                     ;USER TO DIRECT CMS LINKUP
                     ;
                    FILERX:
Ø63D 3EØØ                   MVI     A,      Ø
Ø63F 32Ø3Ø1                 STA     COUNT
Ø642 32Ø4Ø1                 STA     COUNT+1
Ø645 CD1CØ7                 CALL    CPNAME
Ø648 CD7DØ7                 CALL    CRLF
Ø64B CD69Ø6                 CALL    RESTRT          ;SETS UP FILE CONTROL BLOCK
Ø64E CD7DØ7                 CALL    CRLF
Ø651 CD5AØ7                 CALL    MAKE            ;DELETES AND CREATES DISK FILE
Ø654 CDEBØ8                 CALL    BETA            ;PREPARES CMS TO TRANSMIT FILE
```

```
0657 CD7D07            CALL    CRLF
065A CD8F08            CALL    HAUL                ;RECEIVES FILE FROM CMS
065D CDE907            CALL    FILEWR              ;WRITES FILE ON DISK
0660 CD3708            CALL    CLOSE               ;CLOSES DISK FILE
0663 CD4608            CALL    TALLY               ;PRINTS RECORD COUNT;
0666 C32004            JMP     TX                  ;RETURNS TO TRANSMIT MODE

                 ;CLEARS OUT OLD FILE CONTROL BLOCK AND SETS UP NEW ONE
                RESTRT:
0669 11DB02            LXI     D,      MSG3
066C CDAC07            CALL    MESSAGE             ;PROMPTS "FILENAME.FILETYPE"
066F 3E00              MVI     A,      0
0671 32EA05            STA     FCB2                ;PADS NEW FCB
0674 21EB05            LXI     H,      FCB2+1
0677 3E20              MVI     A,      20H         ;BLANK CHAR
0679 060B              MVI     B,      11
                 PAD1:
067B 77                MOV     M,      A
067C 23                INX     H
067D 05                DCR     B
067E C27B06            JNZ     PAD1
0681 3E00              MVI     A,      0
0683 0604              MVI     B,      4
0685 21F605            LXI     H,      FCB2+12
                 PAD2:
0688 77                MOV     M,      A
0689 23                INX     H
068A 05                DCR     B
068B C28806            JNZ     PAD2

068E 0E01              MVI     C,      1
0690 CD0500            CALL    BDOS
0693 FE41              CPI     'A'                  ;ASKS FOR DESIRED DISK AND
                                                   ;NOTIFIES DISK DRIVE
0695 CAA006            JZ      AONE
0698 FE42              CPI     'B'
```

```
Ø69A CAA5Ø6             JZ      BONE
Ø69D C313Ø7             JMP     REPEAT

              AONE:
Ø6AØ 1EØØ               MVI     E,      Ø
Ø6A2 C3AAØ6             JMP     DSK

              BONE:
Ø6A5 1EØ1               MVI     E,      1
Ø6A7 C3AAØ6             JMP     DSK

              DSK:
Ø6AA ØEØE               MVI     C,      14      ;CHANGES DISK DRIVE SELECTION
Ø6AC CDØ5ØØ             CALL    BDOS
Ø6AF ØEØ1               MVI     C,      1
Ø6B1 CDØ5ØØ             CALL    BDOS
Ø6B4 FE3A               CPI     ':'             ;NEXT CHAR MUST BE ":"
Ø6B6 C213Ø7             JNZ     REPEAT          ;IF NOT, START OVER
Ø6B9 Ø6Ø9               MVI     B,      9
Ø6BB 21EBØ5             LXI     H,      FCB2+1
              FNAME:
Ø6BE C5                 PUSH    B
Ø6BF E5                 PUSH    H
Ø6CØ ØEØ1               MVI     C,      1
Ø6C2 CDØ5ØØ             CALL    BDOS
Ø6C5 E1                 POP     H
Ø6C6 C1                 POP     B
Ø6C7 FEØ3               CPI     CNTLC
Ø6C9 CAØØØØ             JZ      ØØ
Ø6CC FEØ4               CPI     CNTLD
Ø6CE CA4FØ7             JZ      DIRECT
Ø6D1 FE15               CPI     CNTLU
Ø6D3 CAØDØ7             JZ      DUMMY
Ø6D6 FE2E               CPI     '.'
Ø6D8 CAE4Ø6             JZ      FTYPE
Ø6DB 77                 MOV     M,      A
```

```
Ø6DC 23                 INX     H
Ø6DD Ø5                 DCR     B                ;IF FILENAME EXCEEDS 8 CHAR,
                                                 ;START OVER
Ø6DE CA13Ø7             JZ      REPEAT
Ø6E1 C3BEØ6             JMP     FNAME

                FTYPE:
Ø6E4 Ø6Ø4               MVI     B,      4
Ø6E6 21F3Ø5             LXI     H,      FCB2+9
                FTYPE1:
Ø6E9 C5                 PUSH    B
Ø6EA E5                 PUSH    H
Ø6EB ØEØ1               MVI     C,      1
Ø6ED CDØ5ØØ             CALL    BDOS
Ø6FØ E1                 POP     H
Ø6F1 C1                 POP     B
Ø6F2 FEØ3               CPI     CNTLC
Ø6F4 CAØØØØ             JZ      ØØ
Ø6F7 FEØ4               CPI     CNTLD
Ø6F9 CA4FØ7             JZ      DIRECT
Ø6FC FE15               CPI     CNTLU
Ø6FE CAØDØ7             JZ      DUMMY
Ø7Ø1 FEØD               CPI     CR
Ø7Ø3 C8                 RZ
Ø7Ø4 77                 MOV     M,      A
Ø7Ø5 23                 INX     H
Ø7Ø6 Ø5                 DCR     B                ;IF FILETYPE EXCEEDS 3 CHAR,
                                                 ;START OVER
Ø7Ø7 CA13Ø7             JZ      REPEAT
Ø7ØA C3E9Ø6             JMP     FTYPE1
                DUMMY:
Ø7ØD CD7DØ7             CALL    CRLF
Ø71Ø C369Ø6             JMP     RESTRT
                REPEAT:
Ø713 11F6Ø2             LXI     D,      MSG4     ;PROMPTS "REPEAT"
Ø716 CDACØ7             CALL    MESSAGE
```

```
Ø719 C369Ø6             JMP     RESTRT            ;START OVER
               CPNAME:
Ø71C 119CØ3             LXI     D,      MSG15     ;PROMPT "CMS FILENAME FILETYPE?"
Ø71F CDACØ7             CALL    MESSAGE
Ø722 1144ØA             LXI     D,      BUFF4Ø
               NAME2:
Ø725 D5                 PUSH    D
Ø726 ØEØ1               MVI     C,      1
Ø728 CDØ5ØØ             CALL    BDOS
Ø72B D1                 POP     D
Ø72C FEØ3               CPI     CNTLC
Ø72E CAØØØØ             JZ      ØØ
Ø731 FEØ4               CPI     CNTLD
Ø733 CA4FØ7             JZ      DIRECT
Ø736 FE15               CPI     CNTLU
Ø738 CA49Ø7             JZ      DUMMY2
Ø73B FEØD               CPI     CR
Ø73D CA45Ø7             JZ      NAME3
Ø74Ø 12                 STAX    D
Ø741 13                 INX     D
Ø742 C325Ø7             JMP     NAME2
               NAME3:
Ø745 3E24               MVI     A,      '$'
Ø747 12                 STAX    D
Ø748 C9                 RET
               DUMMY2:
Ø749 CD7DØ7             CALL    CRLF
Ø74C C31CØ7             JMP     CPNAME
               DIRECT:
Ø74F 31ØDØ4             LXI     SP,     STKBTM
Ø752 3E13               MVI     A,      XOFF
Ø754 CD6FØ5             CALL    SEND
Ø757 C3A9Ø4             JMP     CRCV1
               MAKE:
Ø75A ØE13               MVI     C,      19        ;DELETE ANY OLD DISK FILE HAVING
Ø75C 11EAØ5             LXI     D,      FCB2      ;FILENAME, FILETYPE LISTED IN
```

```
                                                        ;NEW FCB
Ø75F CDØ5ØØ            CALL    BDOS
Ø762 ØE16              MVI     C,       22      ;CREATES NEW FILE NAMED ABOVE
Ø764 11EAØ5            LXI     D,       FCB2
Ø767 CDØ5ØØ            CALL    BDOS
Ø76A FEFF              CPI     255
Ø76C CA74Ø7            JZ      NOROOM           ;ZERO INDICATES FULL DISK
Ø76F AF                XRA     A
Ø77Ø 32ØAØ6            STA     FCB2+32          ;ZEROES FILE RECORD COUNTER
Ø773 C9                RET

              NOROOM:
Ø774 114CØ3            LXI     D,       MSG11   ;PROMPTS "DISK FULL"
Ø777 CDACØ7            CALL    MESSAGE
Ø77A C32ØØ4            JMP     TX

              CRLF:
Ø77D ØEØ2              MVI     C,       2       ;STARTS NEW LINE ON CONSOLE
Ø77F 1EØD              MVI     E,       CR
Ø781 CDØ5ØØ            CALL    BDOS
Ø784 ØEØ2              MVI     C,       2
Ø786 1EØA              MVI     E,       LF
Ø788 CDØ5ØØ            CALL    BDOS
Ø78B C9                RET


              OPEN:
Ø78C 11EAØ5            LXI     D,       FCB2    ;OPENS DISK FILE FOR READING
Ø78F ØEØF              MVI     C,       15
Ø791 CDØ5ØØ            CALL    BDOS
Ø794 FEFF              CPI     255
Ø796 CAA1Ø7            JZ      BADF             ;ZERO INDICATES NO SUCH FILE
Ø799 AF                XRA     A
Ø79A 32ØAØ6            STA     FCB2+32          ;ZEROES FILE RECORD COUNTER
Ø79D CD7DØ7            CALL    CRLF
Ø7AØ C9                RET
```

```
                BADF:
07A1 110703             LXI     D,      MSG5A   ;PROMPTS "FILE NOT FOUND"
07A4 CDAC07             CALL    MESSAGE
07A7 33                 INX     SP              ;ADJUSTS STACK POINTER
07A8 33                 INX     SP
07A9 C32004             JMP     TX              ;RETURNS TO TRANSMIT MODE
                ;PRINTS MESSAGE AT ADDR IN DE ON CONSOLE
               MESSAGE:
07AC 0E09               MVI     C,      9
07AE CD0500             CALL    BDOS
07B1 C9                 RET
                ;READS ENTIRE DISK FILE INTO RAM STARTING AT
                ;BUFF (LIMITED TO 52K BYTES)
               FILERD:
                FILERD0:
07B2 213002             LXI     H,      FLIMIT
07B5 220501             SHLD    FCOUNT
07B8 115A0A             LXI     D,      BUFF
                FILERD1:
07BB D5                 PUSH    D
07BC 0E1A               MVI     C,      26
07BE CD0500             CALL    BDOS            ;CHANGES DMA BUFFER ADDR
07C1 11EA05             LXI     D,      FCB2
07C4 0E14               MVI     C,      20
07C6 CD0500             CALL    BDOS            ;READ FILE RECORD
07C9 D1                 POP     D
07CA F5                 PUSH    PSW
07CB CD7D08             CALL    COUNTER
07CE 218000             LXI     H,      80H     ;INCREMENTS BUFF BY 80H
07D1 19                 DAD     D
07D2 EB                 XCHG
07D3 F1                 POP     PSW
07D4 FE00               CPI     0
07D6 C0                 RNZ                     ;IF NOT ZERO, EOF CONTAINED IN
                                                ;LAST RECORD
07D7 2A0501             LHLD    FCOUNT
```

```
Ø7DA 2B                 DCX     H
Ø7DB 22Ø5Ø1             SHLD    FCOUNT
Ø7DE 7C                 MOV     A,      H
Ø7DF FEØØ               CPI     Ø
Ø7E1 C2BBØ7             JNZ     FILERD1         ;ZERO IF BUFFER EXCEEDED
Ø7E4 13                 INX     D
Ø7E5 3E13               MVI     A,      XOFF    ;TEMPORARY EOF --  WILL TRANSMIT
Ø7E7 12                 STAX    D               ;FIRST 52K BYTES OF FILE, THEN
Ø7E8 C9                 RET                     ;COME BACK TO READ MORE
                ;WRITES DISK FILE BY SAME ALGORITHM AS ABOVE
               FILEWR:
Ø7E9 115AØA             LXI     D,      BUFF
                CONT:
Ø7EC Ø68Ø               MVI     B,      8ØH     ;MUST CHECK EACH RECORD FOR EOF
Ø7EE CD7DØ8             CALL    COUNTER
                INLOOP:
Ø7F1 D5                 PUSH    D
                INLOOP2:
Ø7F2 1A                 LDAX    D
Ø7F3 FE1A               CPI     EOF             ;IF EOF, THIS WILL BE LAST
                                               ;RECORD WRITTEN
Ø7F5 CA1CØ8             JZ      LAST
Ø7F8 13                 INX     D
Ø7F9 Ø5                 DCR     B
Ø7FA C2F2Ø7             JNZ     INLOOP2
Ø7FD D1                 POP     D
Ø7FE D5                 PUSH    D
Ø7FF ØE1A               MVI     C,      26      ;CHANGE DMA BUFFER ADDR
Ø8Ø1 CDØ5ØØ             CALL    BDOS
Ø8Ø4 11EAØ5             LXI     D,      FCB2
Ø8Ø7 ØE15               MVI     C,      21      ;WRITE ONE DISK RECORD
Ø8Ø9 CDØ5ØØ             CALL    BDOS
Ø8ØC D1                 POP     D
Ø8ØD F5                 PUSH    PSW
Ø8ØE 218ØØØ             LXI     H,      8ØH     ;INCREMENT BUFF BY 8ØH
Ø811 19                 DAD     D
```

```
Ø812 EB                 XCHG
Ø813 F1                 POP     PSW
Ø814 FEØ1               CPI     1
Ø816 CA3ØØ8             JZ      ERR1             ;1 INDICATES DISK FULL
Ø819 C3ECØ7             JMP     CONT
               ;WRITE LAST DISK RECORD
               LAST:
Ø81C D1                 POP     D
Ø81D ØE1A               MVI     C,      26
Ø81F CDØ5ØØ             CALL    BDOS
Ø822 11EAØ5             LXI     D,      FCB2
Ø825 ØE15               MVI     C,      21
Ø827 CDØ5ØØ             CALL    BDOS
Ø82A FEØ1               CPI     1
Ø82C CA3ØØ8             JZ      ERR1
Ø82F C9                 RET
                ERR1:
Ø83Ø 1178Ø3             LXI     D,      MSG13    ;PROMPTS "DISK FULL"
Ø833 CDACØ7             CALL    MESSAGE
Ø836 C9                 RET
               ;CLOSES DISK FILE
               CLOSE:
Ø837 11EAØ5             LXI     D,      FCB2
Ø83A ØE1Ø               MVI     C,      16
Ø83C CDØ5ØØ             CALL    BDOS
Ø83F 1128Ø3             LXI     D,      MSG7     ;PROMPTS "TRANSMISSION COMPLETE"
Ø842 CDACØ7             CALL    MESSAGE
Ø845 C9                 RET
               ;PRINTS OUT RECORD COUNT
               TALLY:
Ø846 3AØ3Ø1             LDA     COUNT
Ø849 1F                 RAR
Ø84A 1F                 RAR
Ø84B 1F                 RAR
Ø84C 1F                 RAR
Ø84D E6ØF               ANI     ØFH
```

```
Ø84F C63Ø             ADI     3ØH
Ø851 CD37Ø5           CALL    CONOUT
Ø854 3AØ3Ø1           LDA     COUNT
Ø857 E6ØF             ANI     ØFH
Ø859 C63Ø             ADI     3ØH
Ø85B CD37Ø5           CALL    CONOUT
Ø85E 3AØ4Ø1           LDA     COUNT+1
Ø861 1F               RAR
Ø862 1F               RAR
Ø863 1F               RAR
Ø864 1F               RAR
Ø865 E6ØF             ANI     ØFH
Ø867 C63Ø             ADI     3ØH
Ø869 CD37Ø5           CALL    CONOUT
Ø86C 3AØ4Ø1           LDA     COUNT+1
Ø86F E6ØF             ANI     ØFH
Ø871 C63Ø             ADI     3ØH
Ø873 CD37Ø5           CALL    CONOUT
Ø876 1184Ø3           LXI     D,      MSG14
Ø879 CDACØ7           CALL    MESSAGE
Ø87C C9               RET
                ;KEEPS TRACK OF RECORDS READ/WRITTEN
                COUNTER:
Ø87D 3AØ4Ø1           LDA     COUNT+1
Ø88Ø C6Ø1             ADI     1
Ø882 27               DAA
Ø883 32Ø4Ø1           STA     COUNT+1
Ø886 3AØ3Ø1           LDA     COUNT
Ø889 CEØØ             ACI     Ø
Ø88B 32Ø3Ø1           STA     COUNT
Ø88E C9               RET
                ;RECEIVES WORDS FROM LINE USART AND STORES AT BUFF
                HAUL:
Ø88F 116CØ3           LXI     D,      MSG12   ;PROMPTS "RECEIVING"
Ø892 CDACØ7           CALL    MESSAGE
Ø895 1158ØA           LXI     D,      BUFF-2  ;FIRST TWO WORDS WILL BE CR, LF
```

```
                                                ;DON'T WANT THEM ON DISK
0898 0100D0         LXI     B,      BUFFMAX ;BUFF LIMIT IS 52K BYTES
089B 3E13           MVI     A,      XOFF
089D CD6F05         CALL    SEND            ;TELL CMS TO START SENDING
            ;CHECK USART FOR CHARACTER
            FRX1:
08A0 DB61           IN      61H
08A2 E602           ANI     2
08A4 CAA008         JZ      FRX1
08A7 DB60           IN      60H             ;GET CHAR
08A9 FE11           CPI     XON
08AB CAD508         JZ      MARK            ;IF XON, THIS IS END OF FILE
08AE FE13           CPI     XOFF
08B0 CAA008         JZ      FRX1            ;FILTER OUT XOFF AT END OF LINE
08B3 FE7F           CPI     7FH
08B5 CAA008         JZ      FRX1            ;FILTER OUT "NULLS"
08B8 CD3705         CALL    CONOUT          ;PRINT ON CONSCLE
08BB 12             STAX    D               ;STORE IN BUFFER
08BC 13             INX     D               ;INCREMENT BUFFER ADDR
08BD 0B             DCX     B               ;CHECK BUFFER LIMIT NOT EXCEEDED
08BE 78             MOV     A,      B
08BF FE00           CPI     0
08C1 CAE108         JZ      EXCEED
08C4 CDCA08         CALL    BREAK2
08C7 C3A008         JMP     FRX1            ;LOOP FOREVER
            ;CHECK KEYBOARD FOR INTERRUPT
             ;IF INTERRUPT EXISTS, RESET STACK POINTER
             ;AND JUMP TO DIRECT LINKUP MODE
             ;WHERE INTERRUPT CONDITION WILL BE NOTED
             ;AND A SIGNAL SENT TO CMS
            BREAK2:
08CA DBF7           IN      0F7H
08CC E602           ANI     2
08CE C8             RZ
08CF 310D04         LXI     SP,     STKBTM
08D2 C3A904         JMP     CRCV1
```

```
                    ;MARK END OF FILE WITH "EOF"
                    ;LAST CHARS RECEIVED ARE CR,LF,NULL,R;>
                    ;WANT TO BACK UP TO LAST VALID WORD
                   MARK:
Ø8D5 1A                      LDAX    D
Ø8D6 1B                      DCX     D
Ø8D7 FE52                    CPI     'R'
Ø8D9 C2D5Ø8                  JNZ     MARK
Ø8DC 13                      INX     D
Ø8DD 3E1A                    MVI     A,      EOF
Ø8DF 12                      STAX    D
Ø8EØ C9                      RET
                    EXCEED:
Ø8E1 11B5Ø3                  LXI     D,      MSG17   ;PROMPTS "BUFFER LIMIT EXCEEDED"
Ø8E4 CDACØ7                  CALL    MESSAGE
Ø8E7 3E1A                    MVI     A,      EOF     ;MARKS END OF FILE-REMAINDER OF
                                                    ;FILE IS LOST
Ø8E9 12                      STAX    D
Ø8EA C9                      RET
                   ;SENDS "PRINT" TO CMS
                   BETA:
Ø8EB 1145Ø3                  LXI     D,      MSG1Ø
                    GAMMA:
Ø8EE 1A                      LDAX    D
Ø8EF FE24                    CPI     '$'
Ø8F1 CAFEØ8                  JZ      DELTA
Ø8F4 CD37Ø5                  CALL    CONOUT
Ø8F7 CD6FØ5                  CALL    SEND
Ø8FA 13                      INX     D
Ø8FB C3EEØ8                  JMP     GAMMA
                    ;SENDS "FILENAME FILETYPE" TO CMS
                   DELTA:
Ø8FE 1144ØA                  LXI     D,      BUFF4Ø
                    EPSILON:
Ø9Ø1 1A                      LDAX    D
Ø9Ø2 FE24                    CPI     '$'
```

```
Ø9Ø4 C8                    RZ
Ø9Ø5 CD37Ø5                CALL    CONOUT
Ø9Ø8 CD6FØ5                CALL    SEND
Ø9ØB 13                    INX     D
Ø9ØC C3Ø1Ø9                JMP     EPSILON
                   ;SETS UP CMS TO RECEIVE FILE BY COMMANDING
                   ;"EDIT FILENAME FILETYPE"
                  CMS:
Ø9ØF 11Ø1Ø3                LXI     D,      MSG5
                   CMS2:
Ø912 1A                    LDAX    D
Ø913 FE24                  CPI     '$'
Ø915 CA22Ø9                JZ      CMS3
Ø918 CD37Ø5                CALL    CONOUT
Ø91B CD6FØ5                CALL    SEND
Ø91E 13                    INX     D
Ø91F C312Ø9                JMP     CMS2
                   CMS3:
Ø922 1144ØA                LXI     D,      BUFF4Ø
                   CMS4:
Ø925 1A                    LDAX    D
Ø926 FE24                  CPI     '$'
Ø928 CA35Ø9                JZ      CMS5
Ø92B CD37Ø5                CALL    CONOUT
Ø92E CD6FØ5                CALL    SEND
Ø931 13                    INX     D
Ø932 C325Ø9                JMP     CMS4
                   CMS5:
Ø935 3E13                  MVI     A,      XOFF
Ø937 CD6FØ5                CALL    SEND
Ø93A C9                    RET
                   ;ECHOES CMS ANSWER TO CONSOLE
                  ANS:
Ø93B DB61                  IN      61H
Ø93D E6Ø2                  ANI     2
Ø93F CA3BØ9                JZ      ANS
```

```
0942 DB60            IN      60H
0944 FE11            CPI     XON
0946 C8              RZ
0947 FE13            CPI     XOFF              ;FILTERS OUT XOFF
0949 CA3B09          JZ      ANS
094C CD3705          CALL    CONOUT
094F C33B09          JMP     ANS
                 ;RECEIVES CMS ANSWERS AND ECHOES TO CONSOLE
                 ;FILTERS OUT XOFF,CR,LF,AND >
                ANS2:
0952 DB61            IN      61H
0954 E602            ANI     2
0956 CA5209          JZ      ANS2
0959 DB60            IN      60H
095B FE11            CPI     XON
095D C8              RZ
095E FE13            CPI     XOFF
0960 CA5209          JZ      ANS2
0963 FE0D            CPI     CR
0965 CA5209          JZ      ANS2
0968 FE0A            CPI     LF
096A CA5209          JZ      ANS2
096D FE3E            CPI     '>'
096F CA5209          JZ      ANS2
0972 CD3705          CALL    CONOUT
0975 C35209          JMP     ANS2
                 ;TRANSMITS FILE TO CMS
                 XMIT:
0978 111903          LXI     D,      MSG6      ;PROMPTS "TRANSMITTING"
097B CDAC07          CALL    MESSAGE
097E CD1D0A          CALL    PAUSE             ; DELAY 100 MICROSECS AT
0981 115A0A          LXI     D,      BUFF      ;BEGINNING OF EACH LINE
0984 0E83            MVI     C,      83H       ;132 BYTES
                 XMIT2:
0986 1A              LDAX    D
0987 FE1A            CPI     EOF               ;IF EOF, TRANSMISSION FINISHED
```

```
0989 CAAF09          JZ      XMIT3
098C FE13            CPI     XOFF              ;IF TEMPORARY EOF, MORE DISK
                                               ;FILE REMAINS
098E CAC909          JZ      XMIT4
0991 FE0D            CPI     CR                ;CLOSE OUT LINE AT CARR RETURN
0993 CA080A          JZ      ENDLN
0996 FE0A            CPI     LF                ;FILTER OUT LINEFEEDS
0998 CAA809          JZ      SKIP
099B FE09            CPI     09H               ;CHANGE TAB CHAR TO "?"
099D CC7904          CZ      CHNG1
09A0 47              MOV     B,      A
09A1 CD6F05          CALL    SEND
09A4 0D              DCR     C
09A5 CA0C0A          JZ      ENDLN2            ;IF 132 CHARS EXCEEDED, CMS
                                               ;BUFFER CHOKES
                SKIP:
09A8 13              INX     D
09A9 CDF909          CALL    BREAK3
09AC C38609          JMP     XMIT2
                XMIT3:
09AF 112803          LXI     D,      MSG7      ;PROMPTS "TRANSMISSION COMPLETE"
09B2 CDAC07          CALL    MESSAGE
                XMIT35:
09B5 CD1D0A          CALL    PAUSE
09B8 3E13            MVI     A,      XOFF      ;SENDS DOUBLE XOFF TO SHIFT
09BA CD6F05          CALL    SEND              ;CMS FROM INPUT TO EDIT MODE
09BD CD5209          CALL    ANS2              ;WAIT FOR ANSWER AND DELAY
09C0 CD1D0A          CALL    PAUSE
09C3 3E13            MVI     A,      XOFF
09C5 CD6F05          CALL    SEND
09C8 C9              RET
                ;FOR FILES EXCEEDING 52K, PROGRAM SHIFTS
                ;CMS TO EDIT MODE AND ISSUES "SAVE" COMMAND
                ;AT THIS POINT - CMS SAVES TRANSMITTED DATA
                ;AND RETURNS TO INPUT MODE, AT WHICH TIME
                ;PROGRAM READS NEXT SECTION OF FILE AND TRANSMITS
```

```
                XMIT4:
Ø9C9 CDB5Ø9             CALL    XMIT35
Ø9CC CD3BØ9             CALL    ANS
Ø9CF CD1DØA             CALL    PAUSE
Ø9D2 11F4Ø3             LXI     D,      MSG19
                 XMIT5:
Ø9D5 1A                 LDAX    D
Ø9D6 FE24               CPI     '$'
Ø9D8 CAE5Ø9             JZ      XMIT6
Ø9DB CD37Ø5             CALL    CONOUT
Ø9DE CD6FØ5             CALL    SEND
Ø9E1 13                 INX     D
Ø9E2 C3D5Ø9             JMP     XMIT5
                 XMIT6:
Ø9E5 3E13               MVI     A,      XOFF
Ø9E7 CD6FØ5             CALL    SEND
Ø9EA CD3BØ9             CALL    ANS
Ø9ED 11E8Ø3             LXI     D,      MSG18   ;PROMPTS "RELOADING"
Ø9FØ CDACØ7             CALL    MESSAGE
Ø9F3 CDB2Ø7             CALL    FILERDØ         ;READ NEXT PART OF FILE
Ø9F6 C378Ø9             JMP     XMIT            ;TRANSMIT NEXT PART OF FILE
                 BREAK3:
Ø9F9 DBF7               IN      ØF7H
Ø9FB E6Ø2               ANI     2
Ø9FD C8                 RZ
Ø9FE DBF6               IN      ØF6H
ØAØØ E67F               ANI     7FH
ØAØ2 FEØ4               CPI     CNTLD
ØAØ4 CØ                 RNZ
ØAØ5 C34FØ7             JMP     DIRECT
                 ;SENDS XOFF AFTER EACH LINE
                ENDLN:
ØAØ8 B8                 CMP     B               ;IF LAST CHAR WAS A CR, IGNORE-
ØAØ9 CAA8Ø9             JZ      SKIP            ;CANCELS SKIPPED LINES
                 ENDLN2:
ØAØC 47                 MOV     B,      A
```

```
ØAØD 3E13              MVI     A,      XOFF
ØAØF CD6FØ5            CALL    SEND
ØA12 CD52Ø9            CALL    ANS2
ØA15 CD1DØA            CALL    PAUSE
ØA18 ØE83              MVI     C,      83H     ;132 BYTES
ØA1A C3A6Ø9            JMP     SKIP            ;CONTINUE TRANSMITTING
               ;DELAY APPROX 1ØØ MICROSECONDS
              PAUSE:
ØA1D 21ØØØ2            LXI     H,      2ØØH
               PAUSE2:
ØA2Ø 2B                DCX     H
ØA21 7C                MOV     A,      H
ØA22 FEØØ              CPI     Ø
ØA24 C22ØØA            JNZ     PAUSE2
ØA27 C9                RET
               ;COMMANDS CMS TO "FILE" TRANSMITTED DATA
              FILE:
ØA28 CD1DØA            CALL    PAUSE
ØA2B 114ØØ3            LXI     D,      MSG8
               FILE2:
ØA2E 1A                LDAX    D
ØA2F FE24              CPI     '$'
ØA31 CA3EØA            JZ      FILE3
ØA34 CD37Ø5            CALL    CONOUT
ØA37 CD6FØ5            CALL    SEND
ØA3A 13                INX     D
ØA3B C32EØA            JMP     FILE2
               FILE3:
ØA3E 3E13              MVI     A,      XOFF
ØA4Ø CD6FØ5            CALL    SEND
ØA43 C9                RET
ØA44                   BUFF4Ø: DS      2Ø
ØA58                           DS      2
ØA5A =                 BUFF    EQU     $       ;BUFFER STARTS AT END OF PROGRAM
ØA5A                   END 1ØØH
```

APPENDIX H

DATLINK

```
                        ;UPDATED 14 AUG 78    VERS 73
ORG     100H
JMP     START
BDOS    EQU     5H      ;ENTRY POINT
XON     EQU     11H     ; END OF LINE FROM VIRTUAL MACHINE
XOFF    EQU     13H     ;END OF LINE TO VIRTUAL MACHINE
CR      EQU     0DH     ;CARRIAGE RETURN
LF      EQU     0AH     ;LINE FEED
FF      EQU     0CH     ;FORM FEED
EOF     EQU     1AH     ;END OF FILE CHAR FOR DISK WRITE
BUFF    EQU     880H    ;START OF MEMORY BUFFER
CONV    EQU     0FE0EH  ;MONITOR CONVERSION ROUTINE
RUB     EQU     7FH     ;DELETE CHARACTER
CNTLC   EQU     03H     ;WARM BOOT
CNTLD   EQU     04H     ;RESTORES "DIRECT LINKUP" MODE
CNTLG   EQU     07H     ;PRINT INSTRUCTIONS
CNTLI   EQU     09H     ;TAB CHARACTER
CNTLP   EQU     10H     ;CONTROL P TURNS PRINTER ON AND OFF
CNTLR   EQU     12H     ;CONTROL R FOR RECEIVE FILE
CNTLT   EQU     14H     ;CONTROL T FOR TRANSMIT FILE
CNTLU   EQU     15H     ;DELETE LINE
PPREG:  DS      1       ;PRINTER CONTROL REG;0 OFF,1 ON
```

```
MSG1:   DB      CR,LF,'DIAL 2721 FOR LINE -- CONTROL G FOR INSTRUCTIONS',CR,LF,'$'
MSG2:   DB      '               CONTROL C - REBOOT',CR,LF
        DB      '               CONTROL D - RETURN TO DIRECT LINKUP',CR,LF
        DB      '               CONTROL G - INSTRUCTIONS',CR,LF
        DB      '               CONTROL I - TAB',CR,LF
        DB      '               CONTROL P - PRINTER ON/OFF',CR,LF
        DB      '               CONTROL T - TRANSMIT FILE',CR,LF
        DB      '               CONTROL U - DELETE LINE',CR,LF
        DB      '               RUBOUT    - DELETE CHARACTER',CR,LF
        DB      '               XMIT      - INTERRUPT CMS',CR,LF,'$'
MSG3:   DB      CR,LF,'DISK:FILENAME.FILETYPE',CR,LF,'$'
MSG4:   DB      CR,LF,'REPEAT',CR,LF,'$'
MSG5:   DB      'EDIT $'
MSG5A:  DB      'FILE NOT FOUND',CR,LF,'>$'
MSG6:   DB      'TRANSMITTING',CR,LF,'$'
MSG7:   DB      'TRANSMISSION COMPLETE',CR,LF,'$'
MSG8:   DB      'FILE$'
MSG9:   DB      ' DATA POINTS',CR,LF,'$'
MSG10:  DB      'TO RECEIVE FILE, USE LINK PROGRAM',CR,LF,'$'
MSG11:  DB      ' SCANS PER SECOND',CR,LF,'$'
MSG12:  DB      'RUN CONTROL NUMBER $'
MSG15:  DB      'CMS FILENAME FILETYPE?',CR,LF,'$'
STACK:  DS      20
STKBTM  EQU     $


                START:
 03A0 31A003            LXI     SP,     STKBTM
 03A3 3E00              MVI     A,      0
 03A5 320301            STA     PPREG           ;INITIALLY PRINTER IS OFF
 03A8 110401            LXI     D,      MSG1
 03AB CDF206            CALL    MESSAGE         ;PROMPTS USER TO CALL FOR LINE
 03AE CD4105            CALL    BOARD           ;INITIALIZES SBC 534 BOARD
 03B1 DB60              IN      60H
                ;
                ;TRANSMIT MODE
```

```
                    ;
                    TX:
03B3 DB61                   IN      61H             ;CHECKS LINE FOR MESSAGE
03B5 E602                   ANI     2
03B7 C23C04                 JNZ     CRCV1
03BA DBF7                   IN      0F7H            ;CHECKS KEYBOARD
03BC E602                   ANI     2
03BE CAB303                 JZ      TX              ;LOOPS UNTIL ONE OF THE ABOVE
03C1 0E01                   MVI     C,      1
03C3 CD0500                 CALL    BDOS            ;READ CHAR FROM CONSOLE
03C6 FE0D                   CPI     CR              ;CHECK FOR CR
03C8 CA2504                 JZ      RCV             ;SWITCH TO RECEIVE MODE
03CB FE10                   CPI     CNTLP
03CD CA9604                 JZ      PRTCONT         ;TURN PRINTER ON/OFF
03D0 FE12                   CPI     CNTLR
03D2 CAC805                 JZ      FILERX          ;RECEIVE FILE MODE
03D5 FE14                   CPI     CNTLT
03D7 CA9E05                 JZ      FILETX          ;TRANSMIT FILE MODE
03DA FE03                   CPI     CNTLC
03DC CA0000                 JZ      00H             ;ESCAPE BY REBOOTING
03DF FE07                   CPI     CNTLG           ;PRINT INSTRUCTIONS
03E1 CA7904                 JZ      GOUGE
03E4 FE09                   CPI     CNTLI
03E6 CC1F04                 CZ      CHNG4           ;TRANSMIT TAB CHAR "?"
03E9 FE7F                   CPI     RUB
03EB CC0F04                 CZ      CHNG2           ;TRANSMIT DELETE CHAR SYMBOL "@"
03EE FE15                   CPI     CNTLU
03F0 CA1704                 JZ      CHNG3           ;TRANSMIT DELETE LINE SYMBOL "["
                                                    ;AND XOFF
03F3 4F                     MOV     C,      A
03F4 FE11                   CPI     XON
03F6 CA0504                 JZ      CTX
03F9 3A0301                 LDA     PPREG           ;CHECK IF PRINTER ON
03FC FE00                   CPI     0
03FE CA0504                 JZ      CTX
0401 79                     MOV     A,      C
```

```
0402 CDBF04             CALL    DRIVER
                CTX:
0405 79                 MOV     A,      C
0406 CD0205             CALL    SEND            ;SENDS CHAR TO VIRTUAL MACHINE
0409 C3B303             JMP     TX              ;LOOPS FOREVER
                CHNG1:
040C 3E3F               MVI     A,      '?'
040E C9                 RET
                CHNG2:
040F 3E08               MVI     A,      08H     ;BACKSPACE
0411 CDCA04             CALL    CONOUT
0414 3E40               MVI     A,      '@'
0416 C9                 RET
                CHNG3:
0417 3E5B               MVI     A,      '['
0419 CD0205             CALL    SEND
041C C32504             JMP     RCV
                CHNG4:
041F 3E3F               MVI     A,      '?'
0421 CDCA04             CALL    CONOUT
0424 C9                 RET
                ;
                ;RECEIVE MODE
                ;
                RCV:
0425 3A0301             LDA     PPREG           ;CHECK IF PRINTER ON
0428 FE00               CPI     0
042A CA3704             JZ      CRCV
042D 3E0D               MVI     A,      CR      ;START NEW LINE ON PRINTER
042F CDBF04             CALL    DRIVER
0432 3E0A               MVI     A,      LF
0434 CDBF04             CALL    DRIVER
                CRCV:
0437 3E13               MVI     A,      XOFF    ;END OF LINE CHAR
0439 CD0205             CALL    SEND
                CRCV1:
```

```
                        ;HL REGISTER POINTS TO ADDR FOR NEXT WORD RECEIVED
                        ;DE REGISTER POINTS TO ADDR OF NEXT WORD TO BE PRINTED
043C 218008                     LXI     H,      BUFF    ;FIFO BUFFER ADDR FOR
                                                        ;RECEIVED DATA
043F 118008                     LXI     D,      BUFF
                        RX1:
0442 CD0D05                     CALL    BREAK
0445 DB61                       IN      61H             ;CHECK LINE FOR CHAR
0447 E602                       ANI     02H
0449 CAD504                     JZ      CKPRT           ;IF LINE NOT READY, CHECK IF
                                                        ;BUFFER CAUGHT UP

                        RX:
044C DB60                       IN      60H             ;INPUT WORD FROM LINE
044E E67F                       ANI     7FH
0450 FE11                       CPI     XON
0452 CA5F04                     JZ      CATCH           ;IF END OF LINE, LET BUFFER
                                                        ;CATCH UP
0455 FE13                       CPI     XOFF
0457 CA4204                     JZ      RX1             ;FILTER OUT XOFF CHAR
045A 77                         MOV     M,      A       ;STORE CHAR
045B 23                         INX     H
045C C34204                     JMP     RX1             ;LOOP UNTIL END OF LINE
                        CATCH:
045F 77                         MOV     M,      A       ;STORE LAST WORD

                        LOOOP:
0460 1A                         LDAX    D               ;NEXT WORD TO BE PRINTED
0461 FE11                       CPI     XON
0463 CAB303                     JZ      TX              ;GO BACK TO TRANSMIT MODE
0466 CDCA04                     CALL    CONOUT          ;PRINT ON CONSOLE
0469 3A0301                     LDA     PPREG           ;CHECK IF PRINTER ON
046C FE00                       CPI     0
046E CA7504                     JZ      BACK
0471 1A                         LDAX    D
0472 CDBF04                     CALL    DRIVER
```

```
                    BACK:
Ø475 13                     INX     D
Ø476 C36ØØ4                 JMP     LOOOP               ;LOOP UNTIL CAUGHT UP
                    GOUGE:
Ø479 1139Ø1                 LXI     D,      MSG2
                    GLOOP:
Ø47C 1A                     LDAX    D
Ø47D FE24                   CPI     '$'
Ø47F CAB3Ø3                 JZ      TX
Ø482 CDCAØ4                 CALL    CONOUT
Ø485 47                     MOV     B,      A
Ø486 3AØ3Ø1                 LDA     PPREG
Ø489 FEØØ                   CPI     Ø
Ø48B CA92Ø4                 JZ      GLP
Ø48E 78                     MOV     A,      B
Ø48F CDBFØ4                 CALL    DRIVER
                    GLP:
Ø492 13                     INX     D
Ø493 C37CØ4                 JMP     GLOOP
                    PRTCONT:
Ø496 3AØ3Ø1                 LDA     PPREG               ;CHECK IF PRINTER ON OR OFF
Ø499 FEØØ                   CPI     Ø
Ø49B C2B3Ø4                 JNZ     PRTOFF              ;IF ON, WANT TO TURN OFF
Ø49E CD78Ø5                 CALL    USART2
Ø4A1 3EØ1                   MVI     A,      1
Ø4A3 32Ø3Ø1                 STA     PPREG               ;LATER ROUTINES CHECK THIS ADDR
Ø4A6 3EØD                   MVI     A,      CR          ;START PRINTER ON NEW LINE
Ø4A8 CDBFØ4                 CALL    DRIVER
Ø4AB 3EØA                   MVI     A,      LF
Ø4AD CDBFØ4                 CALL    DRIVER
Ø4BØ C3B3Ø3                 JMP     TX                  ;RETURN TO TRANSMIT MODE
                    PRTOFF:
Ø4B3 3E3Ø                   MVI     A,      3ØH         ;CONTROL WORD TO TURN PRINT OFF
Ø4B5 D363                   OUT     63H
Ø4B7 3EØØ                   MVI     A,      Ø
Ø4B9 32Ø3Ø1                 STA     PPREG               ;LATER ROUTINES CHECK THIS ADDR
```

```
Ø4BC C3B3Ø3                JMP     TX
                  ;ROUTINE TO DRIVE PRINTER USART
                  DRIVER:
Ø4BF F5                    PUSH    PSW
                  SLO:
Ø4CØ DB63                  IN      63H                 ;WAIT UNTIL XMITTER READY
Ø4C2 ØF                    RRC
Ø4C3 D2CØØ4                JNC     SLO
Ø4C6 F1                    POP     PSW
Ø4C7 D362                  OUT     62H
Ø4C9 C9                    RET

                  ;ROUTINE TO DRIVE CONSOLE USART
                  CONOUT:
Ø4CA F5                    PUSH    PSW
                    SLO2:
Ø4CB DBF7                  IN      ØF7H
Ø4CD ØF                    RRC
Ø4CE D2CBØ4                JNC     SLO2
Ø4D1 F1                    POP     PSW
Ø4D2 D3F6                  OUT     ØF6H
Ø4D4 C9                    RET
                  ;KEEPS TRACK OF WHICH DATA HAS BEEN PRINTED
                  CKPRT:
Ø4D5 7D                    MOV     A,      L
Ø4D6 BB                    CMP     E
Ø4D7 CA42Ø4                JZ      RX1                 ;CAUGHT UP - NO NEED TO PROCEED
Ø4DA DBF7                  IN      ØF7H
Ø4DC ØF                    RRC
Ø4DD D242Ø4                JNC     RX1                 ;CONSOLE NOT READY - NO NEED TO
                                                       ;PROCEED
Ø4EØ 3AØ3Ø1                LDA     PPREG               ;CHECK PRINTER ON
Ø4E3 FEØØ                  CPI     Ø
Ø4E5 CAEEØ4                JZ      CKP2                ;PRINTER NOT ON - NO NEED TO
                                                       ;PROCEED
Ø4E8 DB63                  IN      63H
```

```
04EA 0F                 RRC
04EB D24204             JNC     RX1                ;PRINTER NOT READY - NO NEED TO
                                                   ;PROCEED
                CKP2:
04EE 1A                 LDAX    D                  ;NEXT WORD TO BE PRINTED
04EF D3F6               OUT     0F6H               ;OUT TO CONSOLE
04F1 D362               OUT     62H                ;OUT TO PRINTER
04F3 13                 INX     D
04F4 7D                 MOV     A,      L          ;CHECK AGAIN TO SEE IF BUFFER IS
04F5 BB                 CMP     E                  ;CAUGHT UP - IF SO, RESET BUFFER
04F6 C24204             JNZ     RX1
04F9 218008             LXI     H,      BUFF
04FC 118008             LXI     D,      BUFF
04FF C34204             JMP     RX1
                ;DRIVES USART ON HIGH SPEED LINE
                SEND:
0502 F5                 PUSH    PSW
                WAIT:
0503 DB61               IN      61H
0505 0F                 RRC
0506 D20305             JNC     WAIT
0509 F1                 POP     PSW
050A D360               OUT     60H
050C C9                 RET
                ;CHECKS KEYBOARD FOR INTERRUPT
                BREAK:
050D DBF7               IN      0F7H
050F E602               ANI     2
0511 C8                 RZ                         ;IF NONE, GO TO RECEIVE MODE
0512 DBF6               IN      0F6H               ;INTRPT PRESENT, CHECK IF BREAK
0514 E67F               ANI     7FH
0516 FE11               CPI     XON
0518 C0                 RNZ                        ;IGNORE IF NOT BREAK
0519 3E3F               MVI     A,      3FH        ;CONTROL - DRIVES XMIT LINE LOW
051B D361               OUT     61H                ;HOLD LINE LOW FOR 2 WORDLENGTHS
051D 010004             LXI     B,      400H       ;WAIT 10 MILLISECS
```

```
                    DLA1:
0520 ØB                     DCX     B
0521 78                     MOV     A,      B
0522 FEØØ                   CPI     Ø
0524 CA32Ø5                 JZ      DLA3
0527 DB61                   IN      61H             ;CHECK LINE FOR CHAR
0529 E6Ø2                   ANI     2
052B CA2ØØ5                 JZ      DLA1
052E DB6Ø                   IN      6ØH
0530 77                     MOV     M,      A
0531 23                     INX     H
                    DLA3:
0532 Ø1AØØ5                 LXI     B,      5AØH    ;DELAY 16 MILLISEC
                    DLA2:
0535 ØB                     DCX     B
0536 78                     MOV     A,      B
0537 FEØØ                   CPI     Ø
0539 C235Ø5                 JNZ     DLA2
                    RESET:
053C 3E37                   MVI     A,      37H
053E D361                   OUT     61H
0540 C9                     RET

                    BOARD:

;
;
;
;
;   THIS ROUTINE INITIALIZES THE 534 BOARD, THE TIMERS, AND THE TWO USARTS
;   NEEDED TO DRIVE THE IBM HIGH SPEED LINE AND THE MODEL 4Ø PRINTER
;
;   BASE ADDR OF 534 BOARD          6ØH
;   CMD ADDR OF LINE USART          61H
;   DATA ADDR OF LINE USART         6ØH
;   CMD ADDR OF PTR USART           63H
;   DATA ADDR OF PTR USART          62H
```

```
;
;   TWO MORE USARTS AND ONE 8255 PARALLEL INTERFACE AND THEIR TIMERS ARE
;   AVAILABLE ON THE 534 BOARD.  NEW INTERFACES MUST BE PROGRAMMED BEFORE USE
;
;
0541 F3                DI                           ;DISABLES 8080 INTERRUPTS
0542 D36F              OUT     6FH                  ;RESETS BOARD
0544 D36C              OUT     6CH                  ;SELECTS BOARD CONTROL BLOCK
0546 CD4E05            CALL    TIMER                ;INITIALIZE PIT CHIPS
0549 CD6B05            CALL    USART                ;INITIALIZE USARTS FOR IBM LINE AND PTR
054C FB                EI                           ;REENABLES INTERRUPTS
054D C9                RET

                  ;
                  ;
                  ;MUST SET UP TIMER CHIPS ACCORDING TO PAGE 3-12 OF 534 MANUAL
                  ;  CHIP 0 HAS THREE TIMERS ON IT
                  ;  TIMERS 0 AND 1 OF CHIP 0 ARE CONNECTED TO USARTS 1 AND 2
                  ;  RESPECTIVELY, DRIVING THE IBM LINE AND THE PRINTER
                  ;
                  TIMER:
054E D36C              OUT     6CH                  ;SELECT CONTROL BLOCK
0550 3E36              MVI     A,      36H          ;SELECT TIMER 0 FOR LINE USART
0552 D363              OUT     63H                  ;ADDR OF COUNTER 0 MODE CONTROL
0554 3E40              MVI     A,      40H          ;
0556 D360              OUT     60H                  ;SET N-40H IN TIMER 0
0558 3E00              MVI     A,      0H           ;CCLK/N=19.2KHZ FOR 1200 BAUD,
055A D360              OUT     60H                  ;BRF=16X
055C 3E76              MVI     A,      76H          ;
055E D363              OUT     63H                  ;SELECT TIMER 1 FOR PTR USART
0560 3E08              MVI     A,      8H           ;
0562 D361              OUT     61H                  ;SET N=8 IN TIMER 1
0564 3E00              MVI     A,      0H           ;CCLK/N=153.6KHZ FOR 9600 BAUD,
0566 D361              OUT     61H                  ;BRF=16X
0568 D36D              OUT     6DH                  ;PUTS BOARD IN DATA BLOCK
056A C9                RET
```

```
                  ;
                  ;
                  ;SET UP BOTH USARTS WITH RESETS AND MODE WORDS
                  ;
                  USART:
 056B 3ECA               MVI     A,      0CAH    ;2 STOP, PAR DISABLED, 7 BITS
 056D D361               OUT     61H
 056F 3E5A               MVI     A,      5AH     ;1 STOP, PAR DISABLED, 7 BITS
 0571 D363               OUT     63H
 0573 3E37               MVI     A,      37H
 0575 D361               OUT     61H
 0577 C9                 RET
                  USART2:
 0578 3E33               MVI     A,      33H
 057A D363               OUT     63H
 057C C9                 RET
                  ;THIS SECTION PERTAINS TO TRANSFERRING COMPLETE
                  ;FILES BETWEEN MDS AND IBM 360
                  ;
FCB     EQU      5CH                 ;FCB ADDR
FCBCN   EQU      FCB+0               ;DISK NAME
FCBFN   EQU      FCB+1               ;FILENAME(8CHAR)
FCBFT   EQU      FCB+9               ;FILETYPE (3CHAR)
FCBRL   EQU      FCB+12              ;REEL NUMBER
FCBRC   EQU      FCB+15              ;FILE RECORD COUNT (0-127)
FCB2:   DS       33                  ;NEW FILENAME AND FILETYPE
FCBCR   EQU      FCB+32              ;NEXT RECORD NUMBER

                  ;SUBR PROMPTS CONSLE FOR FILE TO BE XMITTED, SETS UP FILE
                  ;CONTROL BLOCK, OPENS NEW CMS FILE, TRANSMITS FILE, AND
                  ;RETURNS USER TO DIRECT CMS LINKUP
                  ;
                 FILETX:
 059E CDD105             CALL    RESTRT          ;SETS UP FILE CONTROL BLOCK
```

```
Ø5A1 CDC7Ø6            CALL    CRLF
Ø5A4 CD89Ø6            CALL    CPNAME          ;CP/CMS FILENAME, FILETYPE
Ø5A7 CDC7Ø6            CALL    CRLF
Ø5AA CDD2Ø6            CALL    OPEN            ;OPENS DISK FILE
Ø5AD CDØ3Ø7            CALL    FILERD          ;READS DISK FILE
Ø5BØ CD22Ø7            CALL    ECHO            ;ECHO FILE INFO
Ø5B3 CD5FØ7            CALL    CMS             ;PREPARES CMS TO RECEIVE FILE
Ø5B6 CD8BØ7            CALL    ANS             ;WAITS FOF ANSWER
Ø5B9 CDC8Ø7            CALL    XMIT            ;TRANSMITS FILE
Ø5BC CD8BØ7            CALL    ANS
Ø5BF CD3CØ8            CALL    FILE            ;"FILES" FILE IN CMS
Ø5C2 CD8BØ7            CALL    ANS
Ø5C5 C3B3Ø3            JMP     TX              ;RETURNS TO TRANSMIT MODE


                 ;THIS PROGRAM DOES NOT HAVE RECEIVE FILE MODE
                FILERX:
Ø5C8 1127Ø3            LXI     D,      MSG1Ø
Ø5CB CDF2Ø6            CALL    MESSAGE
Ø5CE C3B3Ø3            JMP     TX


                 ;CLEARS OUT OLD FILE CONTROL BLOCK AND SETS UP NEW ONE
                 RESTRT:
Ø5D1 11AEØ2            LXI     D,      MSG3
Ø5D4 CDF2Ø6            CALL    MESSAGE         ;PROMPTS "FILENAME.FILETYPE"
Ø5D7 3EØØ              MVI     A,      Ø
                       STA     FCB2            ;PADS NEW FCB
Ø5DC 217EØ5            LXI     H,      FCB2+1
Ø5DF 3E2Ø              MVI     A,      2ØH             ;BLANK CHAR
Ø5E1 Ø6ØB              MVI     B,      11
                 PAD1:
Ø5E3 77                MOV     M,      A
Ø5E4 23                INX     H
Ø5E5 Ø5                DCR     B
Ø5E6 C2E3Ø5            JNZ     PAD1
```

```
05E9 3E00               MVI     A,      0
05EB 0604               MVI     B,      4
05ED 218905             LXI     H,      FCB2+12
                PAD2:
05F0 77                 MOV     M,      A
05F1 23                 INX     H
05F2 05                 DCR     B
05F3 C2F005             JNZ     PAD2

05F6 0E01               MVI     C,      1
05F8 CD0500             CALL    BDOS
05FB FE41               CPI     'A'             ;ASKS FOR DESIRED DISK
                                                ;AND NOTIFIES DISK DRIVE
05FD CA0D06             JZ      AONE
0600 FE42               CPI     'B'
0602 CA1206             JZ      BONE
0605 FE04               CPI     CNTLD
0607 CABC06             JZ      DIRECT
060A C38006             JMP     REPEAT

                AONE:
060D 1E00               MVI     E,      0
060F C31706             JMP     DSK

                BONE:
0612 1E01               MVI     E,      1
0614 C31706             JMP     DSK

                DSK:
0617 0E0E               MVI     C,      14      ;CHANGES DISK DRIVE SELECTION
0619 CD0500             CALL    BDOS
061C 0E01               MVI     C,      1
061E CD0500             CALL    BDOS
0621 FE3A               CPI     ':'             ;NEXT CHAR MUST BE ":"
0623 C28006             JNZ     REPEAT          ;IF NOT, START OVER
0626 0609               MVI     B,      9
```

```
Ø628 217EØ5              LXI     H,      FCB2+1
               FNAME:
Ø62B C5                  PUSH    B
Ø62C E5                  PUSH    H
Ø62D ØEØ1                MVI     C,      1
Ø62F CDØ5ØØ              CALL    BDOS
Ø632 E1                  POP     H
Ø633 C1                  POP     B
Ø634 FEØ3                CPI     CNTLC
Ø636 CAØØØØ              JZ      ØØ
Ø639 FEØ4                CPI     CNTLD
Ø63B CABCØ6              JZ      DIRECT
Ø63E FE15                CPI     CNTLU
Ø64Ø CA7AØ6              JZ      DUMMY
Ø643 FE2E                CPI     '.'
Ø645 CA51Ø6              JZ      FTYPE
Ø648 77                  MOV     M,      A
Ø649 23                  INX     H
Ø64A Ø5                  DCR     B               ;IF FILENAME EXCEEDS 8 CHAR,
Ø64B CA8ØØ6              JZ      REPEAT          ;START OVER
Ø64E C32BØ6              JMP     FNAME

               FTYPE:
Ø651 Ø6Ø4                MVI     B,      4
Ø653 2186Ø5              LXI     H,      FCB2+9
               FTYPE1:
Ø656 C5                  PUSH    B
Ø657 E5                  PUSH    H
Ø658 ØEØ1                MVI     C,      1
Ø65A CDØ5ØØ              CALL    BDOS
Ø65D E1                  POP     H
Ø65E C1                  POP     B
Ø65F FEØ3                CPI     CNTLC
Ø661 CAØØØØ              JZ      ØØ
Ø664 FEØ4                CPI     CNTLD
Ø666 CABCØ6              JZ      DIRECT
```

```
0669 FE15              CPI     CNTLU
066B CA7A06            JZ      DUMMY
066E FE0D              CPI     CR
0670 C8                RZ
0671 77                MOV     M,      A
0672 23                INX     H
0673 05                DCR     B                ;IF FILETYPE EXCEEDS 3 CHAR,
0674 CA8006            JZ      REPEAT           ;START OVER
0677 C35606            JMP     FTYPE1
                DUMMY:
067A CDC706            CALL    CRLF
067D C3D105            JMP     RESTRT
                REPEAT:
0680 11C902            LXI     D,      MSG4     ;PROMPTS "REPEAT"
0683 CDF206            CALL    MESSAGE
0686 C3D105            JMP     RESTRT           ;START OVER
                CPNAME:
0689 117303            LXI     D,      MSG15    ;PROMPTS "CMS FILENAME FILETYPE?"
068C CDF206            CALL    MESSAGE
068F 115808            LXI     D,      BUFF40
                NAME2:
0692 D5                PUSH    D
0693 0E01              MVI     C,      1
0695 CD0500            CALL    BDOS
0698 D1                POP     D
0699 FE03              CPI     CNTLC
069B CA0000            JZ      00
069E FE04              CPI     CNTLD
06A0 CABC06            JZ      DIRECT
06A3 FE15              CPI     CNTLU
06A5 CAB606            JZ      DUMMY2
06A8 FE0D              CPI     CR
06AA CAB206            JZ      NAME3
06AD 12                STAX    D
06AE 13                INX     D
06AF C39206            JMP     NAME2
```

```
                 NAME3:
06B2 3E24               MVI     A,      '$'
06B4 12                 STAX    D
06B5 C9                 RET
                 DUMMY2:
06B6 CDC706             CALL    CRLF
06B9 C38906             JMP     CPNAME
                 DIRECT:
06BC 31A003             LXI     SP,     STKBTM
06BF 3E13               MVI     A,      XOFF
06C1 CD0205             CALL    SEND
06C4 C33C04             JMP     CRCV1
                 CRLF:
06C7 3E0D               MVI     A,      CR
06C9 CDCA04             CALL    CONOUT
06CC 3E0A               MVI     A,      LF
06CE CDCA04             CALL    CONOUT
06D1 C9                 RET


                 OPEN:
06D2 117D05             LXI     D,      FCB2     ;OPENS DISK FILE FOR READING
06D5 0E0F               MVI     C,      15
06D7 CD0500             CALL    BDOS
06DA FEFF               CPI     255
06DC CAE706             JZ      BADF             ;ZERO INDICATES NO SUCH FILE
06DF AF                 XRA     A
06E0 329D05             STA     FCB2+32          ;ZEROES FILE RECORD COUNTER
06E3 CDC706             CALL    CRLF
06E6 C9                 RET
                 BADF:
06E7 11DA02             LXI     D,      MSG5A    ;PROMPTS "FILE NOT FOUND"
06EA CDF206             CALL    MESSAGE
06ED 33                 INX     SP               ;ADJUSTS STACK POINTER
06EE 33                 INX     SP
06EF C3B303             JMP     TX               ;RETURNS TO TRANSMIT MODE
```

```
                MESSAGE:                      ;PRINTS MESSAGE AT ADDR IN DE ON CONSOLE
06F2 0E09               MVI     C,      9
06F4 CD0500             CALL    BDOS
06F7 C9                 RET
                MESS2:
06F8 1A                 LDAX    D
06F9 13                 INX     D
06FA FE24               CPI     '$'
06FC C8                 RZ
06FD CDCA04             CALL    CONOUT
0700 C3F806             JMP     MESS2
                ;READS ENTIRE DISK FILE INTO RAM STARTING AT
                ;BUFF (LIMITED TO 52K BYTES)
                FILERD:
                FILERD0:
0703 118008             LXI     D,      BUFF
                FILERD1:
0706 D5                 PUSH    D
0707 0E1A               MVI     C,      26
0709 CD0500             CALL    BDOS                    ;CHANGES DMA BUFFER ADDR
070C 117D05             LXI     D,      FCB2
070F 0E14               MVI     C,      20
0711 CD0500             CALL    BDOS                    ;READ FILE RECORD
0714 D1                 POP     D
0715 F5                 PUSH    PSW
0716 218000             LXI     H,      80H     ;INCREMENTS BUFF BY 80H
0719 19                 DAD     D
071A EB                 XCHG
071B F1                 POP     PSW
071C FE00               CPI     0
071E C0                 RNZ
071F C30607             JMP     FILERD1
                ;ROUTINE TO ECHO FILE RECORD DATA TO CONSOLE
                ;-FIRST FILE RECORD (BUFF) CONTAINS "DATA1" (ASCII),
                ;START CHAN (HEX), FINAL CHAN (HEX), NUMBER DATA POINTS$
                ;SCAN RATE$RUN CONTROL NUMBER$ (ALL IN ASCII)
```

```
                        ;BUFF+20H CONTAINS UPPER MEMORY LIMIT OF DATA --
                        ;BUFF+30H CONTAINS SCAN WORD LENGTH (EFFECTIVE
                        ;LINE LENGTH FOR TRANSMISSION TO CMS)
                        ;
                        ECHO:
0722 CDC706                     CALL    CRLF                    ;SKIP LINE
0725 118008                     LXI     D,      BUFF            ;FIRST LINE OF FILE
0728 0605                       MVI     B,      6H
                        ELOOP:
072A 1A                         LDAX    D                       ;ECHO 6 DIGIT NAME
072B CDCA04                     CALL    CONOUT
072E 13                         INX     D
072F 05                         DCR     B
0730 C22A07                     JNZ     ELOOP
0733 CDC706                     CALL    CRLF                    ;SKIP LINE
0736 118708                     LXI     D,      BUFF+8H ;NUMBER OF DATA
0739 CDF806                     CALL    MESS2                   ;POINTS LOCATED HERE
073C D5                         PUSH    D
073D 111803                     LXI     D,      MSG9            ;"DATA POINTS"
0740 CDF206                     CALL    MESSAGE
0743 D1                         POP     D                       ;DE REG CONTAINS ADDR OF
0744 CDF806                     CALL    MESS2                   ;FILE PARAMETERS
0747 D5                         PUSH    D
0748 114B03                     LXI     D,      MSG11           ;"SCANS PER SECOND"
074B CDF206                     CALL    MESSAGE
074E 115F03                     LXI     D,      MSG12           ;"RUN CONTROL NUMBER"
0751 CDF206                     CALL    MESSAGE
0754 D1                         POP     D
0755 CDF806                     CALL    MESS2
0758 CDC706                     CALL    CRLF
075B CDC706                     CALL    CRLF
075E C9                         RET
                        ;
                        ;SETS UP CMS TO RECEIVE FILE BY COMMANDING
                        ;"EDIT FILENAME FILETYPE"
                        CMS:
```

```
075F 11D402             LXI     D,      MSG5
                CMS2:
0762 1A                 LDAX    D
0763 FE24               CPI     '$'
0765 CA7207             JZ      CMS3
0768 CDCA04             CALL    CONOUT
076B CD0205             CALL    SEND
076E 13                 INX     D
076F C36207             JMP     CMS2
                CMS3:
0772 115808             LXI     D,      BUFF40
                CMS4:
0775 1A                 LDAX    D
0776 FE24               CPI     '$'
0778 CA8507             JZ      CMS5
077B CDCA04             CALL    CONOUT
077E CD0205             CALL    SEND
0781 13                 INX     D
0782 C37507             JMP     CMS4
                CMS5:
0785 3E13               MVI     A,      XOFF
0787 CD0205             CALL    SEND
078A C9                 RET
                ;ECHOES CMS ANSWER TO CONSOLE
                ANS:
078B DB61               IN      61H
078D E602               ANI     2
078F CA8B07             JZ      ANS
0792 DB60               IN      60H
0794 FE11               CPI     XON
0796 C8                 RZ
0797 FE13               CPI     XOFF            ;FILTERS OUT XOFF
0799 CA8B07             JZ      ANS
079C CDCA04             CALL    CONOUT
079F C38B07             JMP     ANS
                ;RECEIVES CMS ANSWERS AND ECHOES TO CONSOLE
```

```
                  ;FILTERS OUT XOFF,CR,LF,AND >
                  ANS2:
07A2 DB61                 IN      61H
07A4 E602                 ANI     2
07A6 CAA207               JZ      ANS2
07A9 DB60                 IN      60H
07AB FE11                 CPI     XON
07AD C8                   RZ
07AE FE13                 CPI     XOFF
07B0 CAA207               JZ      ANS2
07B3 FE0D                 CPI     CR
07B5 CAA207               JZ      ANS2
07B8 FE0A                 CPI     LF
07BA CAA207               JZ      ANS2
07BD FE3E                 CPI     '>'
07BF CAA207               JZ      ANS2
07C2 CDCA04               CALL    CONOUT
07C5 C3A207               JMP     ANS2
                  ;TRANSMITS FILE TO CMS
                  XMIT:
07C8 11EC02               LXI     D,      MSG6     ;PROMPTS "TRANSMITTING"
07CB CDF206               CALL    MESSAGE
07CE CD3108               CALL    PAUSE            ; DELAY 100 MICROSECS AT BEGIN-
                                                   ;NING OF EACH LINE
07D1 110009               LXI     D,      BUFF+80H
07D4 3AB008               LDA     BUFF+30H         ;NUMBER CHAR PER LINESCAN
07D7 67                   MOV     H,      A
                  XMIT2:
07D8 1A                   LDAX    D
07D9 CDFB07               CALL    ASCII
07DC 78                   MOV     A,      B
07DD CD0205               CALL    SEND
07E0 79                   MOV     A,      C
07E1 CD0205               CALL    SEND
07E4 25                   DCR     H
07E5 CA1808               JZ      ENDLN2
```

```
                SKIP:
07E8 13                 INX     D
07E9 CD0908             CALL    BREAK3
07EC C3D807             JMP     XMIT2
                XMIT3:
07EF 11FB02             LXI     D,      MSG7    ;PROMPTS "TRANSMISSION COMPLETE"
07F2 CDF206             CALL    MESSAGE
                XMIT35:
07F5 3E13               MVI     A,      XOFF    ;SENDS DOUBLE XOFF TO SHIFT
07F7 CD0205             CALL    SEND            ;CMS FROM INPUT TO EDIT MODE
07FA C9                 RET                     ;WAIT FOR ANSWER AND DELAY
                ;ROUTINE CONVERTS HEX BYTE TO TWO ASCII CHARS
                ASCII:
07FB F5                 PUSH    PSW
07FC 0F                 RRC
07FD 0F                 RRC
07FE 0F                 RRC
07FF 0F                 RRC
0800 CD0EFE             CALL    CONV
0803 41                 MOV     B,      C       ;SAVES ONE IN B REG
0804 F1                 POP     PSW
0805 CD0EFE             CALL    CONV            ;OTHER RETURNED IN C REG
0808 C9                 RET
                ;
                BREAK3:
0809 DBF7               IN      0F7H
080B E602               ANI     2
080D C8                 RZ
080E DBF6               IN      0F6H
0810 E67F               ANI     7FH
0812 FE04               CPI     CNTLD
0814 C0                 RNZ
0815 C3BC06             JMP     DIRECT
                ENDLN2:
0818 3E13               MVI     A,      XOFF
081A CD0205             CALL    SEND
```

```
081D CDA207           CALL    ANS2
0820 CD3108           CALL    PAUSE
0823 3AA008           LDA     BUFF+20H          ;SEE IF DATA EXHAUSTED
0826 BA               CMP     D
0827 CAEF07           JZ      XMIT3
082A 3AB008           LDA     BUFF+30H
082D 67               MOV     H,      A
082E C3E807           JMP     SKIP              ;CONTINUE TRANSMITTING
                ;DELAY APPROX 100 MICROSECONDS
                PAUSE:
0831 210002           LXI     H,      200H
                PAUSE2:
0834 2B               DCX     H
0835 7C               MOV     A,      H
0836 FE00             CPI     0
0838 C23408           JNZ     PAUSE2
083B C9               RET
                ;COMMANDS CMS TO "FILE" TRANSMITTED DATA
                FILE:
083C CD3108           CALL    PAUSE
083F 111303           LXI     D,      MSG8
                FILE2:
0842 1A               LDAX    D
0843 FE24             CPI     '$'
0845 CA5208           JZ      FILE3
0848 CDCA04           CALL    CONOUT
084B CD0205           CALL    SEND
084E 13               INX     D
084F C34208           JMP     FILE2
                FILE3:
0852 3E13             MVI     A,      XOFF
0854 CD0205           CALL    SEND
0857 C9               RET

BUFF40:         DS    20
                DS    2
```

APPENDIX I

MODEL 4Ø PRINT PROGRAM

```
;       UPDATED 22ØØ ON 23 MAR 78
;
                            ORG     1ØØH
 Ø1ØØ C372Ø2                JMP     MAIN
BDOS    EQU     5           ;DOS ENTRY
OPENF   EQU     15          ;OPEN FILE
READFR  EQU     2Ø          ;READ FILE RECORD
TYPEC   EQU     Ø2          ;TYPE ON CONSOLE
READC   EQU     Ø1          ;READ FROM CONSOLE
BRKF    EQU     11          ;BREAK KEY FUNCTION
LF      EQU     ØAH         ;LINE FEED
CR      EQU     ØDH         ;CARRIAGE RETURN
FF      EQU     ØCH         ;FORM FEED
TB      EQU     Ø9H
FCB     EQU     5CH         ;FILE CONTROL BLOCK ADDR
BUFF    EQU     8ØH         ;RECORD BUFFER 8ØH-FFH
;FILE CONTROL BLOCK DEFINITIONS
FCBCN   EQU     FCB+Ø       ;DISK NAME
FCBFN   EQU     FCB+1       ;FILE NAME(8 CHAR)
FCBFT   EQU     FCB+9       ;FILE TYPE(3 CHAR)
FCBRL   EQU     FCB+12      ;CURRENT REEL NUMBER
FCBRC   EQU     FCB+15      ;FILE RECORD COUNT(Ø-127)
FCBCR   EQU     FCB+32      ;CURRENT(NEXT) RECORD NUMBER
FCBLN   EQU     FCB+33      ;FCB LENGTH

;VARIABLES
LIMIT1:  DS     2
LIMIT2:  DS     2
LCOUNT:  DS     1           ;LINE COUNTER
CCOUNT:  DS     1           ;CHARACTER COUNTER
PCOUNT:  DS     2           ;PAGE COUNTER
```

```
LNDEX:   DS    1          ;LINE SPACER INDEX
MODE:    DS    1          ;ALL OR PARTIAL MODE INDEX
NEAT:    DS    2          ;INDEX FOR BLANKING FIRST LINE
SKNDEX:  DS    1          ;LINE SKIP INDEX
TEXT:    DS    1          ;1 IF TEXT FILE
TYTLE:   DS    12         ;TITLE WILL BE STORED HERE
STACK:   DS    64         ;RESERVE STACK SPACE
STKBTM   EQU   $

;MESSAGES
$'
MSG15:   DB    'TEXT FILE??  (Y/N) $'
MSG2:    DB    'TYPE 2 FOR DOUBLE SPACE $'
MSG3:    DB    '(DEFAULT = SINGLE SPACE) $'
MSG4:    DB    'FILE NOT FOUND $'
MSG7:    DB    'CHECK FOR ERRORS IN CURRENT RECORD $'
MSG8:    DB    'HAVE A NICE DAY   $'
MSG9:    DB    'DONE $'
MSG10:   DB    'TYPE K TO CANCEL OR SPACE TO CONTINUE $'
MSG13:   DB    'PRINT ALL (A) OR PART (P) ?? $'
MSG14:   DB    'ENTER STRING1,STRING2 -- (LIMIT 15 CHARACTERS EACH) $'

               ;MAIN PROGRAM

 0272 315D01   MAIN:    LXI     SP,     STKBTM
 0275 CDF605            CALL    BOARD
 0278 3E00              MVI     A,      0H
 027A 320C01            STA     MODE
 027D 321001            STA     TEXT
 0280 320701            STA     LCOUNT
               ;OPEN DISK FILE FOR READING
               SETUP:
 0283 115C00            LXI     D,      FCB
 0286 0E0F              MVI     C,      OPENF
 0288 CD0500            CALL    BDOS
               ;CHECK FOR ERRORS
```

```
Ø28B FEFF              CPI     255
Ø28D CA6CØ3            JZ      BADF
                ;GOOD OPEN
Ø29Ø AF                XRA     A
Ø291 327CØØ            STA     FCBCR
Ø294 CDCFØ4            CALL    CRLF
Ø297 1164Ø1            LXI     D,      MSG15
Ø29A CDEDØ4            CALL    CRTMSG
Ø29D ØEØ1              MVI     C,      READC
Ø29F CDØ5ØØ            CALL    BDOS
Ø2A2 321ØØ1            STA     TEXT
Ø2A5 FE59              CPI     'Y'
Ø2A7 CA14Ø3            JZ      FILERD-3
Ø2AA CDCFØ4            CALL    CRLF
Ø2AD 1178Ø1            LXI     D,      MSG2
Ø2BØ CDEDØ4            CALL    CRTMSG
Ø2B3 CDCFØ4            CALL    CRLF
Ø2B6 1191Ø1            LXI     D,      MSG3
Ø2B9 CDEDØ4            CALL    CRTMSG
Ø2BC ØEØ1              MVI     C,      READC
Ø2BE CDØ5ØØ            CALL    BDOS
Ø2C1 FE32              CPI     32H
Ø2C3 CAD3Ø2            JZ      DBL
Ø2C6 3E37              MVI     A,      55
Ø2C8 32ØBØ1            STA     LNDEX
Ø2CB 3EØ6              MVI     A,      6
Ø2CD 32ØFØ1            STA     SKNDEX
Ø2DØ C3DDØ2            JMP     BEGIN


                DBL:
Ø2D3 3E1C              MVI     A,      28
Ø2D5 32ØBØ1            STA     LNDEX
Ø2D8 3EØ3              MVI     A,      3
Ø2DA 32ØFØ1            STA     SKNDEX
```

```
                BEGIN:
02DD AF                 XRA     A
02DE 320701             STA     LCOUNT
02E1 320901             STA     PCOUNT
02E4 320A01             STA     PCOUNT+1
                ;DETERMINE PRINT MODE - ALL OR PARTIAL
02E7 CDCF04             CALL    CRLF
02EA 111F02             LXI     D,      MSG13
02ED CDED04             CALL    CRTMSG
02F0 CDDA04             CALL    RDMSG
02F3 FE50               CPI     'P'
02F5 CC3505             CZ      PART
                ;READ FIRST RECORD
02F8 3A6800             LDA     FCBRL
02FB F5                 PUSH    PSW
02FC 3E24               MVI     A,      '$'
02FE 326800             STA     FCBRL
0301 115D00             LXI     D,      FCBFN
0304 211101             LXI     H,      TYTLE
0307 1A         TITLOOP:
                        LDAX    D
0308 77                 MOV     M,      A
0309 23                 INX     H
030A 13                 INX     D
030B FE24               CPI     '$'
030D C20703             JNZ     TITLOOP
0310 F1                 POP     PSW
0311 326800             STA     FCBRL
0314 11010A             LXI     D,      0A01H
                FILERD:
                        PUSH    D
0318 0E1A               MVI     C,      26
031A CD0500             CALL    BDOS            ;CHANGE DMA BUFFER ADDRESS
```

```
Ø31D 115CØØ             LXI     D,      FCB
Ø32Ø ØE14               MVI     C,      READFR
Ø322 CDØ5ØØ             CALL    BDOS            ;READ FILE RECORD
Ø325 D1                 POP     D
Ø326 F5                 PUSH    PSW
Ø327 218ØØØ             LXI     H,      8ØH
Ø32A 19                 DAD     D
Ø32B EB                 XCHG
Ø32C F1                 POP     PSW
Ø32D FEØØ               CPI     Ø               ;CHECK FOR ERRORS
Ø32F CA17Ø3             JZ      FILERD
Ø332 FEØ1               CPI     Ø1              ;CHECK FOR END OF FILE
Ø334 C4AAØ4             CNZ     ERROR
Ø337 3AØCØ1             LDA     MODE
Ø33A FE2A               CPI     '*'
Ø33C CA71Ø5             JZ      FIND
Ø33F 21ØØØA             LXI     H,      ØAØØH
Ø342 C349Ø3             JMP     NEWPG

                REDY:
Ø345 2AØ3Ø1             LHLD    LIMIT1
Ø348 2B                 DCX     H
                ;ROUTINE STARTS NEW PAGE
                NEWPG:
Ø349 CDF2Ø3             CALL    PLABEL

                ;ROUTINE BEGINS NEW LINE
                NEWLN:
Ø34C 3EØØ               MVI     A,      Ø
Ø34E 32Ø8Ø1             STA     CCOUNT
Ø351 3AØCØ1             LDA     MODE
Ø354 FE2A               CPI     '*'             ;DETERMINE IF IN PARTIAL MODE
Ø356 CCC9Ø3             CZ      CLEAN

                GUTS:
Ø359 CD72Ø3             CALL    GNB             ;STARTS MAIN LOOP
```

```
035C FE0D               CPI     CR
035E CA9103             JZ      ENDLN
0361 FE09               CPI     TB
0363 CADC03             JZ      TAB
0366 CD7A03             CALL    PRCHAR
0369 C35903             JMP     GUTS
                ;END OF MAIN PROGRAM
****************************************************************

                ;SUBROUTINES

                ;BAD OPEN
                BADF:
036C 0601               MVI     B,      01
036E CDAA04             CALL    ERROR
0371 C9                 RET
                GNB:
0372 23                 INX     H
0373 7E                 MOV     A,      M
0374 FE1A               CPI     1AH
0376 CA0005             JZ      DONE
0379 C9                 RET

                ;MAINTAINS CHARACTER COUNT
                PRCHAR:
037A CD7C04             CALL    DRIVER
037D 3A0801             LDA     CCOUNT
0380 3C                 INR     A
0381 320801             STA     CCOUNT
0384 FE73               CPI     115             ; 115 CHARACTERS PER LINE
0386 C0                 RNZ

                TRUNC:
0387 CD7203             CALL    GNB
038A FE0D               CPI     CR
038C CA9103             JZ      ENDLN
```

```
Ø38F 3EØD                MVI     A,      CR

                 ;FINISHES LINE AND CHECKS LINE COUNT
                 ENDLN:
Ø391 CD7CØ4              CALL    DRIVER
Ø394 CD72Ø3              CALL    GNB
Ø397 FEØA                CPI     LF
Ø399 CA9EØ3              JZ      THERE
Ø39C 3EØA                MVI     A,      LF

                 THERE:
Ø39E CD7CØ4              CALL    DRIVER
Ø3A1 CD87Ø4              CALL    BREAK
Ø3A4 3A1ØØ1              LDA     TEXT
Ø3A7 FE59                CPI     'Y'
Ø3A9 CA4CØ3              JZ      NEWLN
Ø3AC 3AØ7Ø1              LDA     LCOUNT
Ø3AF 3C                  INR     A
Ø3BØ 32Ø7Ø1              STA     LCOUNT
Ø3B3 E5                  PUSH    H
Ø3B4 21ØBØ1              LXI     H,      LNDEX
Ø3B7 BE                  CMP     M
Ø3B8 E1                  POP     H
Ø3B9 C24CØ3              JNZ     NEWLN

                 ;OUTPUT FORMFEED TO PRINTER; IF OUT OF PAPER CONDITION
                 ;EXISTS, RECEIPT OF FF TURNS PRINTER OFF.  WHEN IN PARTIAL
                 ;PRINT MODE, THIS SPACES FIRST LINE TO ALIGN DESIRED FIRST
                 ;WORD IN PROPER COLUMN
Ø3E3 3EØC                MVI     A,      FF
Ø3E5 CDA3Ø4              CALL    DRIVER
Ø3E8 3EØØ                MVI     A,      Ø
Ø3EA 32Ø7Ø1              STA     LCOUNT
Ø3C6 C349Ø3              JMP     NEWPG
                 CLEAN:
```

```
Ø3C9 3AØDØ1            LDA     NEAT
Ø3CC 47                MOV     B,      A
              SWEEP:
Ø3CD 3E2Ø              MVI     A,      2ØH
Ø3CF CD7AØ3            CALL    PRCHAR
Ø3D2 Ø5                DCR     B
Ø3D3 C2CDØ3            JNZ     SWEEP
Ø3D6 3EØØ              MVI     A,      Ø
Ø3D8 32ØCØ1            STA     MODE
Ø3DB C9                RET

              ;SKIPS SPACES TO NEXT TAB SETTING
              TAB:
Ø3DC 3AØ8Ø1            LDA     CCOUNT
Ø3DF 47                MOV     B,      A
Ø3EØ E6F8              ANI     ØF8H
Ø3E2 C6Ø8              ADI     Ø8H
Ø3E4 9Ø                SUB     B
Ø3E5 47                MOV     B,      A

              TBLOOP:
Ø3E6 3E2Ø              MVI     A,      2ØH
Ø3E8 CD7AØ3            CALL    PRCHAR
Ø3EB Ø5                DCR     B
Ø3EC C2E6Ø3            JNZ     TBLOOP
Ø3EF C359Ø3            JMP     GUTS

              ;INCREMENTS PAGE NUMBER IN BCD
              PLABEL:
Ø3F2 3A1ØØ1            LDA     TEXT
Ø3F5 FE59              CPI     'Y'
Ø3F7 C8                RZ
Ø3F8 E5                PUSH    H
Ø3F9 3AØFØ1            LDA     SKNDEX
              PGLOOP:
```

```
Ø3FC 47               MOV     B,      A
Ø3FD 3EØA             MVI     A,      LF
Ø3FF CD7CØ4           CALL    DRIVER
Ø4Ø2 Ø5               DCR     B
Ø4Ø3 C2FDØ3           JNZ     PGLOOP+1
Ø4Ø6 115DØ1           LXI     D,      MSG1
Ø4Ø9 CDF5Ø4           CALL    PRMSG
Ø4ØC 16ØØ             MVI     D,      Ø
Ø4ØE 21Ø9Ø1           LXI     H,      PCOUNT
Ø411 7E               MOV     A,      M
Ø412 3C               INR     A
Ø413 27               DAA
Ø414 77               MOV     M,      A
Ø415 23               INX     H
Ø416 7E               MOV     A,      M
Ø417 CEØØ             ACI     Ø
Ø419 77               MOV     M,      A
Ø41A E6FØ             ANI     ØFØH
Ø41C 1F               RAR
Ø41D 1F               RAR
Ø41E 1F               RAR
Ø41F 1F               RAR
Ø42Ø CD51Ø4           CALL    PRPAGE
Ø423 7E               MOV     A,      M
Ø424 E6ØF             ANI     ØFH
Ø426 CD51Ø4           CALL    PRPAGE
Ø429 2B               DCX     H
Ø42A 7E               MOV     A,      M
Ø42B E6FØ             ANI     ØFØH
Ø42D 1F               RAR
Ø42E 1F               RAR
Ø42F 1F               RAR
Ø43Ø 1F               RAR
Ø431 CD51Ø4           CALL    PRPAGE
Ø434 7E               MOV     A,      M
Ø435 E6ØF             ANI     ØFH
```

```
0437 CD5104              CALL    PRPAGE
043A 061E                MVI     B,      30
               LOOPER:
043C 3E20                MVI     A,      20H
043E CD7C04              CALL    DRIVER
0441 05                  DCR     B
0442 C23C04              JNZ     LOOPER
0445 211101              LXI     H,      TYTLE
0448 EB                  XCHG
0449 CDF504              CALL    PRMSG
044C CD6404              CALL    PCR2LF
044F E1                  POP     H
0450 C9                  RET

               ;PRINTS PAGE NUMBER DIGIT
               PRPAGE:
0451 C630                ADI     30H
0453 FE30                CPI     30H
0455 C25E04              JNZ     PRPG
0458 47                  MOV     B,      A
0459 7A                  MOV     A,      D
045A FE01                CPI     01
045C C0                  RNZ
045D 78                  MOV     A,      B
               PRPG:
045E 1601                MVI     D,      01
0460 CD7C04              CALL    DRIVER
0463 C9                  RET
               ;PRINTER FORMAT CONTROL
               PCR2LF:
0464 3E0D                MVI     A,      CR
0466 CD7C04              CALL    DRIVER
0469 3E0A                MVI     A,      LF
046B CD7C04              CALL    DRIVER
046E 3E0A                MVI     A,      LF
0470 CD7C04              CALL    DRIVER
```

```
0473 3A0701          LDA     LCOUNT
0476 C603            ADI     03
0478 320701          STA     LCOUNT
047B C9              RET

             ;CHECKS STATUS AND XMITS DATA TO USART
             DRIVER:
047C F5              PUSH    PSW
             STS:
047D DB63            IN      63H
047F 0F              RRC
0480 D27D04          JNC     STS
0483 F1              POP     PSW
0484 D362            OUT     62H
0486 C9              RET


             ;CHECK BREAK KEY (ANY KEY) FOR INTERRUPT
             BREAK:
0487 0E0B            MVI     C,      BRKF
0489 E5              PUSH    H
048A CD0500          CALL    BDOS
048D E1              POP     H
048E 0F              RRC
048F D0              RNC
0490 E5              PUSH    H
0491 CDCF04          CALL    CRLF
0494 11F801          LXI     D,      MSG10
0497 CDED04          CALL    CRTMSG
049A CDCF04          CALL    CRLF
049D CDDA04          CALL    RDMSG           ;EMPTY UART BUFFER
04A0 CDDA04          CALL    RDMSG           ;WAIT FOR NEXT CHAR
04A3 FE4B            CPI     'K'
04A5 E1              POP     H
04A6 C0              RNZ
04A7 C30005          JMP     DONE
```

```
                  ;PRINT ERROR MESSAGE ON CONSOLE
                  ERROR:
04AA E5                     PUSH    H
04AB CDCF04                 CALL    CRLF
04AE 3E07                   MVI     A,      07
04B0 CDE204                 CALL    WRMSG
04B3 78                     MOV     A,      B
04B4 FE01                   CPI     01
04B6 CABE04                 JZ      ERR1
04B9 FE03                   CPI     03
04BB CAC704                 JZ      ERR3

                  ERR1:
04BE 11AB01                 LXI     D,      MSG4    ;FILE NOT FOUND
04C1 CDED04                 CALL    CRTMSG
04C4 C30005                 JMP     DONE

                  ERR3:
04C7 11BB01                 LXI     D,      MSG7
04CA CDED04                 CALL    CRTMSG
04CD E1                     POP     H
04CE C9                     RET

                  ;CARRIAGE RETURN AND LINE FEED
                  CRLF:
04CF 3E0D                   MVI     A,      CR
04D1 CDE204                 CALL    WRMSG
04D4 3E0A                   MVI     A,      LF
04D6 CDE204                 CALL    WRMSG
04D9 C9                     RET

                  ;READ CHARACTER FROM CONSOLE
                  RDMSG:
04DA 0E01                   MVI     C,      READC
```

```
04DC D5                 PUSH    D
04DD CD0500             CALL    BDOS
04E0 D1                 POP     D
04E1 C9                 RET

              ;WRITE CHARACTER TO CONSOLE
              WRMSG:
04E2 C5                 PUSH    B
04E3 D5                 PUSH    D
04E4 0E02               MVI     C,      TYPEC
04E6 5F                 MOV     E,      A
04E7 CD0500             CALL    BDOS
04EA D1                 POP     D
04EB C1                 POP     B
04EC C9                 RET
              ;PRINTS MESSAGE ON CONSOLE
              CRTMSG:
04ED 0E09               MVI     C,      9
04EF E5                 PUSH    H
04F0 CD0500             CALL    BDOS
04F3 E1                 POP     H
04F4 C9                 RET

              ;PRINTS MESSAGE ON PRINTER
              PRMSG:
04F5 1A                 LDAX    D
04F6 FE24               CPI     '$'
04F8 C8                 RZ
04F9 CD7C04             CALL    DRIVER
04FC 13                 INX     D
04FD C3F504             JMP     PRMSG

              ;SIGN OFF ON PRINTER
              DONE:
0500 CD6404             CALL    PCR2LF
0503 3A0B01             LDA     LNDEX
```

```
0506 D603                SUI     3H
0508 2A0701              LHLD    LCOUNT
050B BE                  CMP     M
050C FA1D05              JM      FINISH
050F 3A1001              LDA     TEXT
0512 FE59                CPI     'Y'
0514 CA1D05              JZ      FINISH
0517 11DF01              LXI     D,      MSG8
051A CDF504              CALL    PRMSG
               FINISH:
051D CD6404              CALL    PCR2LF
0520 3E0C                MVI     A,      FF
0522 CD7C04              CALL    DRIVER
0525 3E50                MVI     A,      50H
0527 D363                OUT     063H
0529 CDCF04              CALL    CRLF
052C 11F201              LXI     D,      MSG9
052F CDED04              CALL    CRTMSG
0532 C30000              JMP     0000H


               ;SET UP TO PRINT PART OF PROGRAM
               PART:
0535 CDCF04              CALL    CRLF
0538 3E2A                MVI     A,      '*'
053A 320C01              STA     MODE
053D 113D02              LXI     D,      MSG14
0540 CDED04              CALL    CRTMSG
0543 CDCF04              CALL    CRLF
0546 110009              LXI     D,      900H

0549 13        ;READ AND STORE STRING CHARACTERS-
               STR1:
                         INX     D
054A CDDA04              CALL    RDMSG               ;STRING1 BEGINS AT 901H
                                                     ;STRING2 BEGINS AT 911H
```

```
                                                        ;DELIMITER IS CHARACTER 13H
054D FE7F                  CPI     7FH
054F CAE405                JZ      UNDO1                ;IF RUBOUT SELECTED, CORRECT IT
0552 12                    STAX    D
0553 FE2C                  CPI     ','
0555 C24905                JNZ     STR1
0558 3E13                  MVI     A,      13H
055A 12                    STAX    D
055B 111009                LXI     D,      910H

                   STR2:
055E 13                    INX     D
055F CDDA04                CALL    RDMSG
0562 FE7F                  CPI     7FH
0564 CAED05                JZ      UNDO2
0567 12                    STAX    D
0568 FE0D                  CPI     CR
056A C25E05                JNZ     STR2
056D 3E13                  MVI     A,      13H
056F 12                    STAX    D
0570 C9                    RET

                   FIND:
0571 21010A                LXI     H,      0A01H    ;FIND 1ST STRING AND APPEND ALL
0574 220301                SHLD    LIMIT1
0577 2B                    DCX     H
                   RESET:
0578 110109                LXI     D,      901H     ;AFTER TO TPA STARTING AT 0A01
057B 1A                    LDAX    D
057C FE13                  CPI     13H
057E CAB205                JZ      FIND28

                   ;LOCATE 1ST CHARACTER OF 1ST STRING
                   FIND1:
0581 23                    INX     H
0582 BE                    CMP     M
```

```
0583 C28105             JNZ     FIND1
0586 220301             SHLD    LIMIT1

                ;AFTER 1ST CHARACTER FOUND, CHECK ADDITIONAL CHARACTERS
                ;UNTIL STRING IS EXHAUSTED
                NCR:
05A8 13                 INX     D
05A9 23                 INX     H
058B 1A                 LDAX    D
058C FE13               CPI     13H
058E CA9E05             JZ      FIND2
0591 BE                 CMP     M
0592 C29805             JNZ     FIND15          ;IF NOT CORRECT STRING
                                                ;BEGIN SEARCH AGAIN
0595 C38905             JMP     NCR
               FIND15:
0598 2A0301             LHLD    LIMIT1
059B C37805             JMP     RESET

                FIND2:
059E E5                 PUSH    H               ;SET UP SPACING FOR 1ST LINE-
059F 2A0301             LHLD    LIMIT1          ;DESIRE FIRST WORD TO PRINT IN
                                                ;PROPER COLUMN
05A2 5D                 MOV     E,      L
05A3 3E0A               MVI     A,      LF

                FORMAT:
05A5 2B                 DCX     H
05A6 BE                 CMP     M
05A7 C2A505             JNZ     FORMAT
05AA 7B                 MOV     A,      E
05AB 95                 SUB     L
05AC D601               SUI     1
05AE 320D01             STA     NEAT
05B1 E1                 POP     H
```

```
                    FIND28:
Ø5B2 111109                 LXI     D,      911H

                    ;SEARCH FOR 1ST CHARACTER OF 2ND STRING
                    FIND3:
Ø5B5 1A                     LDAX    D
Ø5B6 FE13                   CPI     13H
Ø5B8 CA45Ø3                 JZ      REDY
Ø5BB BE                     CMP     M
Ø5BC 23                     INX     H
Ø5BD C2BBØ5                 JNZ     FIND3+6
Ø5CØ 2B                     DCX     H
Ø5C1 22Ø5Ø1                 SHLD    LIMIT2          ;SAVE ADDRESS IN CASE THIS IS
                                                    ;CORRECT STRING
Ø5C4 23                     INX     H

                    NCR2:
Ø5C5 13                     INX     D               ;CHECK SUBSEQUENT CHARACTERS
Ø5C6 1A                     LDAX    D               ;IF INCORRECT, BEGIN AGAIN
Ø5C7 FE13                   CPI     13H
Ø5C9 CADBØ5                 JZ      FOUND
Ø5CC BE                     CMP     M
Ø5CD 23                     INX     H
Ø5CE C2D4Ø5                 JNZ     FIND25
Ø5D1 C3C5Ø5                 JMP     NCR2
                    FIND25:
Ø5D4 2AØ5Ø1                 LHLD    LIMIT2
Ø5D7 23                     INX     H
Ø5D8 C3B2Ø5                 JMP     FIND28


                    FOUND:
Ø5DB 2AØ5Ø1                 LHLD    LIMIT2
Ø5DE 3E1A                   MVI     A,      1AH     ;USE CHARACTER 1AH AS DELIMITER
Ø5EØ 77                     MOV     M,      A       ;TO APPENDED MEMORY DATA
Ø5E1 C345Ø3                 JMP     REDY
```

```
                 UNDO1:
05E4 1B                   DCX     D
05E5 1A                   LDAX    D
05E6 CDE204               CALL    WRMSG
05E9 1B                   DCX     D
05EA C34905               JMP     STR1
                 UNDO2:
05ED 1B                   DCX     D
05EE 1A                   LDAX    D
05EF CDE204               CALL    WRMSG
05F2 1B                   DCX     D
05F3 C35E05               JMP     STR2


                 BOARD:
;
;
;   THIS ROUTINE INITIALIZES THE 534 BOARD, THE TIMERS, AND THE TWO USARTS
;   NEEDED TO DRIVE THE IBM HIGH SPEED LINE AND THE MODEL 40 PRINTER
;
;   BASE ADDR OF 534 BOARD          60H
;   CMD ADDR OF LINE USART          61H
;   DATA ADDR OF LINE USART         60H
;   CMD ADDR OF PTR USART           63H
;   DATA ADDR OF PTR USART          62H
;
;   TWO MORE USARTS AND ONE 8255 PARALLEL INTERFACE AND THEIR TIMERS ARE
;   AVAILABLE ON THE 534 BOARD.  NEW INTERFACES MUST BE PROGRAMMED BEFORE USE
;
;
05F6 F3                   DI                        ;DISABLES 8080 INTERRUPTS
05F7 D36F                 OUT     6FH               ;RESETS BOARD
05F9 D36C                 OUT     6CH               ;SELECTS BOARD CONTROL BLOCK
05FB CD0306               CALL    TIMER             ;INITIALIZE PIT CHIPS
```

```
Ø5FE CD14Ø6              CALL    USART                ;INITIALIZE USARTS
Ø6Ø1 FB                  EI                           ;REENABLES INTERRUPTS
Ø6Ø2 C9                  RET

;
;
;MUST SET UP TIMER CHIPS ACCORDING TO PAGE 3-12 OF 534 MANUAL
;  CHIP Ø HAS THREE TIMERS ON IT
;  TIMERS Ø AND 1 OF CHIP Ø ARE CONNECTED TO USARTS 1 AND 2
;  RESPECTIVELY, DRIVING THE IBM LINE AND THE PRINTER
;
                TIMER:
Ø6Ø3 D36C                OUT     6CH                  ;SELECT CONTROL BLOCK
Ø6Ø5 3E76                MVI     A,      76H          ;
Ø6Ø7 D363                OUT     63H                  ;SELECT TIMER 1 FOR PTR USART
Ø6Ø9 3EØ8                MVI     A,      8H           ;
Ø6ØB D361                OUT     61H                  ;SET N=8 IN TIMER 1
Ø6ØD 3EØØ                MVI     A,      ØH           ;CCLK/N=153.6KHZ FOR 96ØØ BAUD,
Ø6ØF D361                OUT     61H                  ;BRF=16X
Ø611 D36D                OUT     6DH                  ;PUTS BOARD IN DATA BLOCK
Ø613 C9                  RET


;
;
;SET UP BOTH USARTS WITH RESETS AND MODE WORDS
;
               USART:
Ø614 3E5A                MVI     A,      5AH          ;1 STOP, PAR DISABLED, 7 BITS
Ø616 D363                OUT     63H
Ø618 3E33                MVI     A,      33H
Ø61A D363                OUT     63H
Ø61C C9                  RET

**********************************************************************
```

APPENDIX J

G02 ASSEMBLY PROGRAM

```
                       1 AUG 1978
;**
;**MDS 8080 PROGRAM INTERFACES DATEL ST-800 ANALOG
;**TO DIGITAL CONVERTER BOARD AND INTEL DYNAMIC
;**MEMORY ACCESS CONTROLLER FOR HIGH SPEED DATA
;**ACQUISITION ---
;**MAXIMUM OF 16 CHANNELS ARE INPUT, CONVERTED,
;**AND STORED IN MEMORY AT A RATE OF 40 KHZ ---
;**PROGRAMMABLE INTERRUPT CONTROLLER AND
;**INTERVAL TIMERS ON THE INTEL SBC 534 BOARD ARE
;**INTERFACED TO PROVIDE VARIABLE SCAN RATES
;**OF ONE TO 2000 SCANS PER SECOND *********
;*****************************************
ORG 100H
 0100 C39C05            JMP     START
                ;
                ;EQUATES
                ;
CR      EQU     0DH     ;CARRIAGE RETURN
LF      EQU     0AH     ;LINE FEED
BDOS    EQU     5H      ;BDOS ENTRY POINT
DMACMD   EQU    17H     ;DMA COMMAND WORD
REVRT   EQU     20H     ;CPU INTERRUPT CLEAR COMMAND
R04     EQU     20H     ;RESTART 04 ADDRESS
R05     EQU     28H     ;RESTART 05 ADDRESS
DMA     EQU     40H     ;DMA BASE ADDRESS
SBC     EQU     60H     ;SBC 534 BASE ADDRESS
JUMP    EQU     0C3H    ;JUMP INSTRUCTION
MASK    EQU     0FCH    ;MASK ALTERATION PORT
```

```
MEMORY EQU      ØAØØH   ;DATA MEMORY BUFFER ADDRESS
                ;
                ;
                ;DATA SAVES
                ;
WCNT:   DS      1H      ;WORD LENGTH SETTING ( X 2 )
ACHAN:  DS      1H      ;START CHANNEL
BCHAN:  DS      1H      ;FINAL CHANNEL
INTVL4: DS      2H      ;TIMER 4 SETTING
INTVL5: DS      2H      ;TIMER 5 SETTING
RCOUNT:         DS      1H      ;SCAN RATE REGISTER
PCOUNT: DS      1H      ;# DATA POINTS REGISTER
LIMIT:  DS      1H      ;MSB OF UPPER MEMORY LIMIT
FLNAME: DB      Ø,'DATA1   XXX',Ø,Ø,Ø,Ø
        DS      17D
STACK:  DS      2ØH     ;SAVE ROOM FOR STACK
STKBTM  EQU     $       ;INITIATE STACK POINTER HERE
                ;
                ;
                ;MESSAGES
                ;
                ;
MSG1:   DB      CR,LF,'ENTER STARTING CHANNEL $'
MSG2:   DB      CR,LF,'ENTER FINAL CHANNEL $'
MSG3:   DB      CR,LF,'CARRIAGE RETURN TO BEGIN $'
MSG4:   DB      CR,LF,'TRY AGAIN, TURKEY $'
MSG5:   DB      CR,LF,'ENTER DESIRED NUMBER OF DATA POINTS '
        DB      CR,LF,LF,'          ENTER     DATA POINTS     DISK SPACE'
        DB      CR,LF,LF,'            A            1024            2K'
        DB      CR,LF,'             B            4096            4K'
        DB      CR,LF,'             C            5120           1ØK'
        DB      CR,LF,'             D           1Ø24Ø           2ØK'
        DB      CR,LF,'             E           26112           52K'
        DB      CR,LF,'$'
MSG6:   DB      CR,LF,'      SELECT SCAN RATE',CR,LF,LF
        DB      '            ENTER        SCANS/SEC        MAX CHANNELS',CR,LF
```

```
        DB      '           A           5000            1',CR,LF
        DB      '           B           4800            2',CR,LF
        DB      '           C           2800            8',CR,LF
        DB      '           D           2200           12',CR,LF
        DB      '           E           1800           16',CR,LF
        DB      '           F           1000           16',CR,LF
        DB      '           G            100           16',CR,LF
        DB      '           H             10           16',CR,LF
        DB      '           I              1           16',CR,LF
        DB      '$'
MSG7:   DB      CR,LF,'WRITE DATA FILE ON DISK?? (Y/N) $'
MSG8:   DB      CR,LF,'ANOTHER DATA RUN DESIRED?? (Y/N) $'
MSG9:   DB      CR,LF,'DISK FULL - TRY ANOTHER - RETURN WHEN READY $'
MSG10:   DB     CR,LF,'DISK WRITE ERROR - TRY ANOTHER - RETURN WHEN READY$'
                ;
                ;
                ;
                ;
                START:
 059C 314E01            LXI     SP,     STKBTM  ;SET UP STACK POINTER
 059F 3EC3              MVI     A,      JUMP    ;JUMP INSTRUCTION
 05A1 322000            STA     R04             ;SET UP INTERRUPT
 05A4 322800            STA     R05             ;JUMP VECTORS
 05A7 21D906            LXI     H,      RESET4  ;ADDR OF INT 4 ROUTINE
 05AA 222100            SHLD    R04+1
 05AD 211B07            LXI     H,      RESET5  ;ADDR OF INT 5 ROUTINE
 05B0 222900            SHLD    R05+1
                ;
                ;CHANGE CPU MASK TO ACCEPT RST 04 AND RST 05 INTERRUPTS
                ;
 05B3 3E4E              MVI     A,      4EH     ;ALLOWS RST 0,4,5,7
 05B5 D3FC              OUT     MASK
                ;
                ;GET VALUES FOR INITIAL AND FINAL CHANNELS AND WORD LENGTH
                ;
                SETUP:
```

```
05B7 CD4307          CALL    DIGIT1       ;GETS CHANNEL VALUES
05BA 210401          LXI     H,      ACHAN ;INITIAL CHANNEL VALUE
05BD 3A0501          LDA     BCHAN        ;FINAL CHANNEL VALUE
05C0 96              SUB     M            ;DETERMINE DIFFERENCE
05C1 F2CA05          JP      DIFF
05C4 CDA907          CALL    OOPS         ;FINAL CAN'T BE LESS
05C7 C3B705          JMP     SETUP        ;BACKUP AND TRY AGAIN
                ;WORD LENGTH IS (DIFFERENCE + 1) X 2
                ;
                DIFF:
05CA C601            ADI     1H
05CC 17              RAL                  ;TIMES 2
05CD 320301          STA     WCNT
                ;
                ;
                ;DETERMINE NUMBER OF DATA POINTS DESIRED
                ;
05D0 11B001          LXI     D,      MSG5 ;PROMPT USER
05D3 0E09            MVI     C,      9H
05D5 CD0500          CALL    BDOS
05D8 CDA307          CALL    KEY
05DB 320B01          STA     PCOUNT       ;SAVE FOR FUTURE USE
                ;
                ;SEE WHICH CHOICE
                ;
                POINT:
05DE FE41            CPI     'A'          ;SEE IF A ENTERED
05E0 CAFD05          JZ      APOINT
05E3 FE42            CPI     'B'          ;SEE IF B ENTERED
05E5 CA0206          JZ      BPOINT
05E8 FE43            CPI     'C'          ;SEE IF C ENTERED
05EA CA0706          JZ      CPOINT
05ED FE44            CPI     'D'          ;SEE IF D ENTERED
05EF CA0C06          JZ      DPOINT
05F2 FE45            CPI     'E'          ;SEE IF E ENTERED
05F4 CA1106          JZ      EPOINT
```

```
05F7 CDA907             CALL    OOPS                ;NOTHING ELSE IS VALID
05FA C3DE05             JMP     POINT
                ;
                APOINT:
05FD 3E0E               MVI     A,      0EH
05FF C31306             JMP     DOWN
                BPOINT:
0602 3E1A               MVI     A,      1AH
0604 C31306             JMP     DOWN
                CPOINT:
0607 3E32               MVI     A,      32H
0609 C31306             JMP     DOWN
                DPOINT:
060C 3E5A               MVI     A,      5AH
060E C31306             JMP     DOWN
                EPOINT:
0611 3ED8               MVI     A,      0D8H
                DOWN:
0613 320C01             STA     LIMIT
                ;
                ;LIMIT IS NOW SET UP
                ;NEXT DETERMINE DESIRED SCAN RATE
                ;
                RATE:
                ;
0616 11E602             LXI     D,      MSG6
0619 0E09               MVI     C,      9H
061B CD0500             CALL    BDOS                ;PROMPT USER
061E CDA307             CALL    KEY                 ;GET USER'S CHOICE OF RATES
0621 110100             LXI     D,      1H          ;LOAD D FOR LATER USE
0624 320A01             STA     RCOUNT              ;SAVE FOR FUTURE USE
0627 FE41               CPI     'A'                 ;SEE IF A ENTERED
0629 CA5A06             JZ      ARATE
062C FE42               CPI     'B'                 ;SEE IF B ENTERED
062E CA6006             JZ      BRATE
0631 FE43               CPI     'C'                 ;SEE IF C ENTERED
```

```
Ø633 CA66Ø6          JZ     CRATE
Ø636 FE44            CPI    'D'              ;SEE IF D ENTERED
Ø638 CA6CØ6          JZ     DRATE
Ø63B FE45            CPI    'E'              ;SEE IF E ENTERED
Ø63D CA72Ø6          JZ     ERATE
Ø64Ø FE46            CPI    'F'              ;SEE IF F ENTERED
Ø642 CA78Ø6          JZ     FRATE
Ø645 FE47            CPI    'G'              ;SEE IF G ENTERED
Ø647 CA7EØ6          JZ     GRATE
Ø64A FE48            CPI    'H'              ;SEE IF H ENTERED
Ø64C CA84Ø6          JZ     HRATE
Ø64F FE49            CPI    'I'              ;SEE IF I ENTERED
Ø651 CA8DØ6          JZ     IRATE
Ø654 CDA9Ø7          CALL   OOPS             ;MUST BE A - I TO BE VALID
Ø657 C316Ø6          JMP    RATE
                 ;
                 ;SET UP REGISTERS ACCORDINGLY
                 ;
                 ARATE:
Ø65A 217AØØ          LXI    H,      7AH      ;COUNTS ARE DETERMINED IN
Ø65D C393Ø6          JMP    RASET            ;THE FOLLOWING MANNER:
                 BRATE:
Ø66Ø 2187ØØ          LXI    H,      87H      ;        1.2288 MHZ
Ø663 C393Ø6          JMP    RASET            ;  N  =  ----------
                 CRATE:                      ;        SCANS/SEC
Ø666 213EØ1          LXI    H,      13EH     ;
Ø669 C393Ø6          JMP    RASET            ;
                 DRATE:                      ;SINCE THE SERVICE ROUTINE
Ø66C 21B6Ø1          LXI    H,      1B6H     ;REQUIRES 98.5 MICROSECONDS
Ø66F C393Ø6          JMP    RASET            ;TO COMPLETE, DETERMINE THE
                 ERATE:                      ;SETTING BEING ENTERED HERE
Ø672 2132Ø2          LXI    H,      232H     ;BY THE FORMULA
Ø675 C393Ø6          JMP    RASET            ;
                 FRATE:                      ;
Ø678 2154Ø4          LXI    H,      454H     ;  N* = N - 1.2288 X 98.5
Ø67B C393Ø6          JMP    RASET            ;
```

```
                GRATE:                          ;
067E 21872F             LXI     H,      2F87H   ;
0681 C39306             JMP     RASET           ;NORMALLY THE SETTING IN TIMER 5
                HRATE:                          ;IS 1, BUT FOR INTERVALS OVER
0684 21C4EF             LXI     H,      0EFC4H  ;50 MILLISECONDS, THE COUNT N
0687 110200             LXI     D,      2H      ;MUST BE DIVIDED BY SOME
068A C39306             JMP     RASET           ;NUMBER M TO REDUCE THE COUNT
                IRATE:                          ;BELOW 0FFFFH;THEN THE TIMER 5
068D 21FAEF             LXI     H,      0EFFAH  ;SETTING INCREASED ACCORDINGLY
0690 111400             LXI     D,      14H
                RASET:                          ;LOAD REGISTERS
0693 220601             SHLD    INTVL4          ;TIMER 4 SETTING
0696 EB                 XCHG
0697 220801             SHLD    INTVL5          ;TIMER 5 SETTING
                ;
                ;RATE OF SCAN IS NOW SET INTO EFFECT
                ;
                ;
                BEGIN:
                ;
                ;NOW READY TO BEGIN SCANNING WHEN PROMPTED
069A 117F01             LXI     D,      MSG3
069D 0E09               MVI     C,      9H
069F CD0500             CALL    BDOS
06A2 CDA307             CALL    KEY
                ;
                ;
06A5 110000             LXI     D,      0H      ;ZERO DE REG
06A8 3A0301             LDA     WCNT
06AB 5F                 MOV     E,      A       ;VALUE OF LENGTH REG
                ;
                ;SET UP DMA AND ST-800 BOARDS
                ;
06AC CDBE06             CALL    DMASET
06AF CDLD06             CALL    TIMSET
                ;
```

```
                          ;DMA AND TIMER NOW SET AND RUNNING -
                          ;NOTHING TO DO BUT WAIT
                          ;
                          WAIT:
06B2 AF                           XRA     A
06B3 C3B206                       JMP WAIT
                          ;
                          ;END OF MAIN PROGRAM
                          ;
                          ;
                          ;
                          ;
                          ;
                          ;
                          ;
                          ;SUBROUTINES
                          ;************************************************
                          ;
                          ;
                          ;*ROUTINE TO INITIALIZE AND RESET DMA AND ST-800
                          ; BOARDS -
                          ;*ST-800 IS ADDRESSED VIA DMA BOARD
                          ;*DMA IS SET UP TO GENERATE A LEVEL 4 INTERRUPT
                          ; WHENEVER ONE SCAN IS COMPLETED -
                          ;**********ADDRESS LISTING FOLLOWS**********
                          ;       DMA BASE ADDR              40H
                          ;       OUTPORT0/INPORT0           40H
                          ;       OUTPORT1/INPORT1           41H
                          ;       OUTPORT2                   42H
                          ;       DMA STATUS                 46H
                          ;       DMA RESET                  49H
                          ;       DMA COMMAND                4AH
                          ;       LENGTH REGISTER (LSB)      4CH
                          ;       LENGTH REGISTER (MSB)      4DH
                          ;       MEMORY ADDR REG (LSB)      4EH
                          ;       MEMORY ADDR REG (MSB)      4FH
```

```
                ;
                ;
                DMASET:
06B6 D349               OUT     DMA+9H              ;RESET DMA
06B8 3A0301             LDA     WCNT                ;LSB OF LENGTH REG
06BB D34C               OUT     DMA+0CH
06BD AF                 XRA     A                   ;MSB IS ZERO
06BE D34D               OUT     DMA+0DH
06C0 21000A             LXI     H,      MEMORY
06C3 7D                 MOV     A,      L           ;LSB OF MEMORY ADDR
06C4 D34E               OUT     DMA+0EH
06C6 7C                 MOV     A,      H           ;MSB OF MEMORY ADDR
06C7 D34F               OUT     DMA+0FH
06C9 3A0401             LDA     ACHAN               ;STARTING CHANNEL
06CC D340               OUT     DMA
06CE 3A0501             LDA     BCHAN               ;FINAL CHANNEL
06D1 D341               OUT     DMA+1H
06D3 3E17               MVI     A,      DMACMD      ;ENABLES INTERRUPT, 8
06D5 D34A               OUT     DMA+0AH             ;BIT XFER TO MEMORY
06D7 FB                 EI                          ;ENABLE INTERRUPTS
06D8 C9                 RET
                ;
                ;DMA NOW READY TO GO WHEN COMMAND WORD IS ISSUED
                ;
                RESET4:
06D9 D349               OUT     DMA+9H              ;RESET DMA
06DB 3A0301             LDA     WCNT                ;LENGTH REG SETTING
06DE D34C               OUT     DMA+0CH
06E0 AF                 XRA     A                   ;MSB OF LENGTH REG IS 0
06E1 D34D               OUT     DMA+0DH
06E3 3E20               MVI     A,      REVRT       ;CLEARS INT 4 FROM CPU
06E5 D3FD               OUT     0FDH                ;INTERRUPT PENDING STACK
06E7 3E17               MVI     A,      DMACMD      ;COMMAND BYTE
06E9 D34A               OUT     DMA+0AH
06EB FB                 EI                          ;REENABLES INTERRUPTS
06EC C9                 RET                         ;DMA IS READY TO GO
```

```
                    ;
                    ;
                    ;
                    ;*ROUTINE TO INITIALIZE INTERRUPT CONTROLLER AND CASCADED
                    ; INTERVAL TIMERS 4 AND 5 ON SBC 534 BOARD --
                    ;**********ADDRESS LISTING FOLLOWS**********
                    ;      SBC BASE ADDR           60H
                    ;      TIMER 4 ADDR            65H
                    ;      TIMER 5 ADDR            66H
                    ;      SELECT TIMER ADDR       67H
                    ;      PIC1 COMMAND ADDR       6AH
                    ;      PIC1 COMMAND ADDR       6BH
                    ;      SELECT CONTROL BLOCK    6CH
                    ;      SELECT DATA BLOCK       6DH
                    ;      RESET SBC BOARD         6FH
                    ;
                    ;
                    TIMSET:
                    ;
                    ;SET UP PROGRAMMABLE INTERRUPT CONTROLLER
                    ;
06ED D36D                  OUT     SBC+0DH             ;SELECT DATA BLOCK
06EF 3E16                  MVI     A,      16H         ;1ST BYTE OF PIC1 CMD
06F1 D36A                  OUT     SBC+0AH
06F3 AF                    XRA     A                   ;2ND BYTE IS ZERO
06F4 D36B                  OUT     SBC+0BH
06F6 3EFD                  MVI     A,      0FDH        ;MASK ALL INTERRUPTS TO
06F8 D36B                  OUT     SBC+0BH             ;PIC1 EXCEPT IR1
                    ;
                    ;SET UP PROGRAMMABLE INTERVAL TIMERS
                    ;
06FA D36C                  OUT     SBC+0CH             ;SELECT CONTROL BLOCK
06FC 3EB0                  MVI     A,      0B0H        ;SELECT TIMER 5 AS
06FE D367                  OUT     SBC+7H              ;INTERVAL TIMER
0700 3A0801                LDA     INTVL5              ;LSB OF TIMER 5 COUNT
0703 D366                  OUT     SBC+6H
```

```
0705 3A0901        LDA    INTVL5+1         ;MSB OF TIMER 5 COUNT
0708 D366          OUT    SBC+6H
070A 3E76          MVI    A,      76H      ;SELECT TIMER 4 AS CLOCK
070C D367          OUT    SBC+7H  ;FOR TIMER 5
070E 3A0601        LDA    INTVL4           ;LSB OF TIMER 4 COUNT
0711 D365          OUT    SBC+5H
0713 3A0701        LDA    INTVL4+1         ;MSB OF TIMER 4 COUNT
0716 D365          OUT    SBC+05H
                 ;
                 ;INTERRUPT TIMER IS NOW SET AND RUNNING
                 ;
0718 D342          OUT    DMA+2H  ;DMA "GO" INSTRUCTION
                 ;
                 ;DMA IS NOW SET AND RUNNING
                 ;
071A C9            RET
                 ;
                 ;
                 ;
                 ;ROUTINE TO SERVICE INTERRUPT 5 FROM INTERRUPT TIMER
                 ;
                 RESET5:
071B 3E76          MVI    A,      76H      ;STOPS TIMER 4
071D D367          OUT    SBC+7H
071F 3A0801        LDA    INTVL5           ;RESET LSB OF TIMER 5
0722 D366          OUT    SBC+6H           ;(REMOVES INT 4 FROM BUS)
0724 3A0901        LDA    INTVL5+1         ;RESET MSB OF TIMER 5
0727 D366          OUT    SBC+6H
0729 3E20          MVI    A,      REVRT    ;RESETS CPU
072B D3FD          OUT    0FDH
072D FB            EI                      ;REENABLES INTERRUPTS
                 ;
                 ;NEED TO KEEP TRACK OF MEMORY AREA USED TO PREVENT OVER
                 ;RUNNING LIMIT
                 ;
072E 19            DAD    D                ;DE REG CONTAINS WORDLENGTH
```

```
072F 3A0C01          LDA     LIMIT
0732 BC              CMP     H
0733 CAB207          JZ      DONE               ;EXIT PROGRAM
                ;
                ;IF MEMORY SPACE OKAY, RESET TIMER AND CONTINUE
                ;
0736 3A0601          LDA     INTVL4             ;RESET LSB OF TIMER 4
0739 D365            OUT     SBC+5H
073B 3A0701          LDA     INTVL4+1           ;RESET MSB OF TIMER 4
073E D365            OUT     SBC+5H
                ;
                ;INTERRUPT TIMERS RUNNING AGAIN
                ;
0740 D342            OUT     DMA+2H
                ;
                ;DMA RUNNING AGAIN
                ;
0742 C9              RET
                ;
                ;
                ;
                ;
                ;ROUTINE TO READ IN INITIAL AND FINAL CHANNELS
                ;
                DIGIT1:
0743 114E01          LXI     D,      MSG1
0746 0E09            MVI     C,      9
0748 CD0500          CALL    BDOS               ;PROMPT USER
074B CDA307          CALL    KEY                ;GET ENTERED CHARACTER
074E FE0D            CPI     CR
0750 CA4307          JZ      DIGIT1
0753 D630            SUI     30H                ;REDUCE ASCII
0755 320401          STA     ACHAN
0758 CDA307          CALL    KEY                ;SEE IF SECOND CHAR
075B FE0D            CPI     CR
075D CA7507          JZ      DIGIT2
```

```
0760 D630            SUI    30H              ;REDUCE ASCII
0762 C61A            ADI    1AH              ;CONVERT TO HEX
0764 320401          STA    ACHAN
0767 CDA307          CALL   KEY              ;STILL NEED CR
076A FE0D            CPI    CR
076C CA7507          JZ     DIGIT2
076F CDA907          CALL   OOPS             ;TOO MANY CHARACTERS
0772 C34307          JMP    DIGIT1           ;TRY AGAIN
                 ;
                 ;
                 DIGIT2:
0775 116801          LXI    D,     MSG2
0778 0E09            MVI    C,     9
077A CD0500          CALL   BDOS             ;PROMPT USER
077D CDA307          CALL   KEY              ;GET CHARACTER
0780 FE0D            CPI    CR
0782 CA7507          JZ     DIGIT2           ;CR NOT ALLOWED YET
0785 D630            SUI    30H
0787 320501          STA    BCHAN
078A CDA307          CALL   KEY              ;GET NEXT CHAR
078D FE0D            CPI    CR
078F C8              RZ                      ;FINISHED IF CR
0790 D630            SUI    30H
0792 C61A            ADI    1AH              ;CONVERT TO HEX
0794 320501          STA    BCHAN
0797 CDA307          CALL   KEY
079A FE0D            CPI    CR
079C C8              RZ                      ;FINISHED IF CR
079D CDA907          CALL   OOPS             ;TOO MANY CHARACTERS
07A0 C37507          JMP    DIGIT2
                 ;
                 ;
                 ;
                 ;ROUTINE TO RETRIEVE CHARACTER FROM KEYBOARD
                 ;
                 KEY:
```

```
 07A3 0E01            MVI     C,      1H
 07A5 CD0500          CALL    BDOS
 07A8 C9              RET
                 ;
                 ;
                 ;ROUTINE PRINTS MESSAGE IF TOO MANY CHARACTERS
                 ;
                 OOPS:
 07A9 119B01          LXI     D,      MSG4
 07AC 0E09            MVI     C,      9
 07AE CD0500          CALL    BDOS
 07B1 C9              RET
                 ;
                 ;
                 DONE:
 07B2 F1              POP     PSW             ;DUMMY POP
 07B3 11F104          LXI     D,      MSG7    ;SEE IF USER WANTS
 07B6 0E09            MVI     C,      9H      ;FILE WRITTEN
 07B8 CD0500          CALL    BDOS
 07BB CDA307          CALL    KEY             ;CHECK ANSWER
 07BE FE4E            CPI     'N'             ;IF NO, CONTINUE
 07C0 CAC907          JZ      GETMOR
U07C3 CD0000          CALL    CRLF
 07C6 C3E607          JMP     FLFILE          ;IF YES, GO WRITE
                 ;
                 ;
                 GETMOR:
 07C9 111405          LXI     D,      MSG8    ;SEE IF USER WANTS
 07CC 0E09            MVI     C,      9H      ;ANOTHER RUN
 07CE CD0500          CALL    BDOS
 07D1 CDA307          CALL    KEY             ;CHECK ANSWER
 07D4 FE59            CPI     'Y'             ;IF YES, GO BACK
 07D6 CADC07          JZ      RERUN
                 ;
                 ;OTHERWISE, ITS TIME TO QUIT
                 ;
```

```
                EXIT:
07D9 C30000             JMP     0H                  ;WARM BOOT
                ;
                ;SET UP FOR ANOTHER RUN
                ;
                RERUN:
07DC 3A1201             LDA     FLNAME+5            ;INCREMENT FILE NAME
07DF 3C                 INR     A
07E0 321201             STA     FLNAME+5
07E3 C39A06             JMP     BEGIN
                ;
                ;NEXT ROUTINE CREATES AND WRITES A DISK FILE -
                ;THE FIRST FILE RECORD CONTAINS INFORMATION
                ;WHICH WILL FACILITATE LATER RETRIEVAL OF THE
                ;DATA ---
                ;THE FIRST FILE RECORD CONTAINS THE DATA FILE
                ;NAME, FIRST CHANNEL, FINAL CHANNEL, SCAN RATE
                ;CODE LETTER, AND DATA POINTS CODE LETTER ---
                ;THE REMAINDER OF THE FIRST FILE RECORD IS ZEROES
                ;
                ;
                FLFILE:
                ;
                ;CREATE FILE ON DISK
                ;
07E6 0E13               MVI     C,      19
07E8 110D01             LXI     D,      FLNAME
07EB CD0500             CALL    BDOS                ;DELETE OLD FILE, SAME NAME
07EE 0E16               MVI     C,      22
07F0 110D01             LXI     D,      FLNAME
07F3 CD0500             CALL    BDOS                ;CREATE NEW FILE
07F6 FEFF               CPI     255                 ;RETURNS 255 IF NOT
07F8 CA6908             JZ      NOROOM              ;ENOUGH DISK SPACE
07FB AF                 XRA     A                   ;ZERO IT
07FC 322D01             STA     FLNAME+32           ;NEXT RECORD COUNT
                ;
```

```
                   ;NEXT SET UP FIRST FILE RECORD
                   ;
                   RECORD:
07FF 3E00                  MVI      A,       0H
0801 118009                LXI      D,       MEMORY-80H
0804 0680                  MVI      B,       80H
                   RLOOP:                              ;ZERO OUT RECORD
0806 12                    STAX     D
0807 13                    INX      D
0808 05                    DCR      B
0809 C20608                JNZ      RLOOP
                   ;
                   ;FILL IN FILE RECORD DATA
                   ;
080C 010E01                LXI      B,       FLNAME+1
080F 118009                LXI      D,       MEMORY-80H
0812 2605                  MVI      H,       5H
                   RLOOP2:
0814 0A                    LDAX     B                  ;COPY FIRST 5 LETTERS
0815 12                    STAX     D                  ;OF FILE NAME INTO
0816 03                    INX      B                  ;RECORD
0817 13                    INX      D
0818 25                    DCR      H
0819 C21408                JNZ      RLOOP2
                   ;
081C 3A0401                LDA      ACHAN              ;FIRST CHANNEL
081F 12                    STAX     D
0820 13                    INX      D
0821 3A0501                LDA      BCHAN              ;FINAL CHANNEL
0824 12                    STAX     D
0825 13                    INX      D
0826 3A0A01                LDA      RCOUNT             ;SCAN RATE CODE
0829 12                    STAX     D
082A 13                    INX      D
082B 3A0B01                LDA      PCOUNT             ;DATA POINT CODE
082E 12                    STAX     D
```

```
                   ;
                   ;FIRST FILE RECORD NOW CONTAINS APPROPRIATE INFORMATION
                   ;
                   ;
                   ;SINCE DMA PUT PAIRS OF DATA BYTES INTO MEMORY IN REVERSE
                   ;ORDER, WANT TO REVERSE THEM BEFORE WRITING ON DISK
                   ;
                   FLIP:
Ø82F 3AØCØ1             LDA     LIMIT               ;UPPER LIMIT ON MEMORY USED
Ø832 21ØØØA             LXI     H,      MEMORY      ;BEGINNING OF DATA
                   ;
                   FLOP:
Ø835 46                 MOV     B,      M           ;GET LSB
Ø836 23                 INX     H
Ø837 4E                 MOV     C,      M           ;GET MSB
Ø838 7Ø                 MOV     M,      B           ;PUT LSB
Ø839 2B                 DCX     H
Ø83A 71                 MOV     M,      C           ;PUT MSB
Ø83B 23                 INX     H
Ø83C 23                 INX     H
Ø83D BC                 CMP     H                   ;CHECK AGAINST LIMIT
Ø83E C235Ø8             JNZ     FLOP
                   ;
                   ;DATA PAIRS NOW IN CORRECT ORDER
                   ;
                   ;
                        ;READY TO START WRITING ONTO DISK
                   ;
                   FWRITE:

Ø841 118ØØ9             LXI     D,      MEMORY-8ØH       ;INFO RECORD
                   ;
                   FLOOP:
Ø844 D5                 PUSH    D                        ;SAVE POINTER
Ø845 ØE1A               MVI     C,      26               ;
Ø847 CDØ5ØØ             CALL    BDOS                     ;CHANGE BUFFER ADDRESS
```

```
Ø84A 11ØDØ1           LXI     D,      FLNAME
Ø84D ØE15             MVI     C,      21
Ø84F CDØ5ØØ           CALL    BDOS                    ;WRITE ONE RECORD
Ø852 D1               POP     D                       ;RETRIEVE POINTER
Ø853 F5               PUSH    PSW                     ;WILL CHECK LATER
Ø854 218ØØØ           LXI     H,      8ØH
Ø857 19               DAD     D                       ;INCREMENT POINTER
Ø858 EB               XCHG                            ;BY 8ØH
Ø859 F1               POP     PSW
Ø85A FEØØ             CPI     ØH                      ;CHECK FOR WRITE ERRORS
Ø85C C277Ø8           JNZ     ERROR
Ø85F 3AØCØ1           LDA     LIMIT                   ;CHECK END OF DATA
Ø862 BA               CMP     D                       ;MSB ONLY
Ø863 CA85Ø8           JZ      CLOSE
Ø866 C344Ø8           JMP     FLOOP                   ;GO DO ANOTHER RECORD
                 ;
                 ;THIS CONTINUES UNTIL ALL DATA WRITTEN ONTO DISK
                 ;
                 ;ROUTINE INFORMS USER THAT DISK OR DIRECTORY IS FULL
                 ;
                 ;
                 NOROOM:
Ø869 1138Ø5           LXI     D,      MSG9
Ø86C ØEØ9             MVI     C,      9H
Ø86E CDØ5ØØ           CALL    BDOS
Ø871 CDA3Ø7           CALL    KEY                     ;WAIT FOR RESPONSE
Ø874 C3E6Ø7           JMP     FLFILE                  ;TRY ANOTHER WRITE
                 ;
                 ;
                 ;
                 ERROR:
Ø877 1167Ø5           LXI     D,      MSG1Ø
Ø87A ØEØ9             MVI     C,      9H
Ø87C CDØ5ØØ           CALL    BDOS                    ;INFO USER OF ERROR
Ø87F CDA3Ø7           CALL    KEY                     ;CHECK FOR RESPONSE
Ø882 C3E6Ø7           JMP     FLFILE                  ;
```

```
                    ;
                    ;IF ERROR OCCURRED IN WRITING ON DISK, ANOTHER WRITE SHOULD
                    ;BE ATTEMPTED ON ANOTHER DISK
                    ;
                    ;
                    ;WHENEVER DATA WRITE IS COMPLETED, NEED TO CLOSE FILE
                    ;
                    CLOSE:
0885 110D01                 LXI     D,      FLNAME
0888 0E10                   MVI     C,      16
088A CD0500                 CALL    BDOS
088D C3C907                 JMP     GETMOR                  ;CHECK WITH USER
                    ;
                    ;
                    ;
                    ;
                    ;
                    ;
                    ;************************************************
0890                END 100H
```

APPENDIX K

PATCH FOR CP/M BIOS PROGRAM

```
                        ;PATCH TO CP/M BIOS PROGRAM
                        ;
                        ;ALTERS JUMP VECTOR BY READDRESSING JUMPS TO
                        ;THE LIST OUT (LO) DEVICE.
                        ;JUMP VECTOR INSTEAD POINTS TO ALTERNATE ROUTINE
                        ;WHICH SENDS CHARACTER TO MODEL 40 PRINTER.
                        ;PRINTER MUST HAVE BEEN PREVIOUSLY SET UP
                        ;BY AN INDEPENDENT ROUTINE (ON.COM)
                        ;
BE00 C344BE                     JMP     BOOT
BE03 C354BE                     JMP     WBOOT
BE06 C3F2BE                     JMP     CONST
BE09 C3F5BE                     JMP     CONIN
BE0C C3FBBE                     JMP     CONOUT
BE0F C3E7BF                     JMP     PATCH
BE12 C301BF                     JMP     PUNCH
BE15 C304BF                     JMP     READER
BE18 C307BF                     JMP     HOME
BE1B C30CBF                     JMP     SELDSK
BFE7                            ORG     0BFE7H
                        PATCH:
BFE7 DB63                       IN      63H     ;CHECK USART STATUS
BFE9 E601                       ANI     1
BFEB CAE7BF                     JZ      PATCH
BFEE 79                         MOV     A,C     ;PUT BYTE IN ACCUM
BFEF D362                       OUT     62H     ;SEND TO USART
BFF1 C9                         RET
```

APPENDIX L

ON ASSEMBLY PROGRAM

```
                          ;THIS ROUTINE INITIALIZES THE INTEL SBC 534 BOARD,
                          ;THE TIMER, AND THE USART NEEDED TO DRIVE THE
                          ;MODEL 40 PRINTER
                          ;
0100                      ORG     100H

                          ;BASE ADDR OF 534 BOARD         60H
                          ;CMD ADDR OF PRINTER USART      63H
                          ;DATA ADDR OF PRINTER USART     62H
                          ;
                 START:
0100 310002               LXI     SP,     200H    ;SET UP STACK
0103 D36F                 OUT     6FH             ;RESETS 534 BOARD
0105 D36C                 OUT     6CH             ;SELECTS CONTROL BLOCK

                 TIMER:
0107 3E76                 MVI     A,      76H     ;SELECT TIMER 1 FOR
0109 D363                 OUT     63H             ;PRINTER USART --
010B 3E08                 MVI     A,      8H      ;SET N=8 IN TIMER 1
010D D361                 OUT     61H             ;CCLK/N = 153.6KHZ FOR 9600
010F 3E00                 MVI     A,      0H      ;BAUD, BRF = 16X
0111 D361                 OUT     61H

                 USART:
0113 D36D                 OUT     6DH             ;SELECT DATA BLOCK

                          ;MODE WORD - SETS UP 1 STOP BIT, ODD PARITY
                          ;ENABLED, 7 BIT WORD, AND A BAUD RATE
                          ;FACTOR OF 16X
```

```
                        ;
0115 3E5A               MVI     A,      5AH     ;MODE WORD
0117 D363               OUT     63H             ;COMMAND PORT

                        ;COMMAND WORD - SETS RTS, ERROR RESET, DTR,
                        ;AND XMIT ENABLE
                        ;
0119 3E33               MVI     A,      33H     ;COMMAND WORD
011B D363               OUT     63H             ;COMMAND PORT

011D C30000             JMP     0H              ;SOFT BOOT

0120                    END     100H
```

APPENDIX M

REDUCE FORTRAN PROGRAM

```
C ** FOURIER COEFFICIENT DETERMINATION **
C    * PROGRAM INPUT CONSISTS OF CHANNELS "J1" TO "JMAX" OF
C    DISCRETIZED DATA USING A COMMON TIME BASE FOR THE SAMPLINGS.
C    * PROGRAM OUTPUT CONSISTS OF FOURIER COEFFICIENTS FOR THE
C    VARIOUS CHANNELS, INCLUDING OPTIONS FOR HIGHER HARMONICS.
C    RELATIVE PHASING BETWEEN THE CHANNELS IS OBTAINED.
C
      DIMENSION Y(5),RMS(5),A(5,5),B(5,5),C(5,5),PHI(5,5),IX(5,500)
     1,X(5,500)
    1 FORMAT (1H0,'ENTER DISK FILE NUMBER (I2)'/)
    2 FORMAT (I2)
    3 FORMAT (1H0,'ENTER FILE NO. (I2), NUMBER OF CHANNELS (I2), SCAN')
    4 FORMAT (1H ,'RATE (I5), FUNDAMENTAL FREQUENCY (F6.0), NUMBER')
    5 FORMAT (1H ,'OF DATA POINTS (I5), COORDINATION NUMBER (I8)'//)
    6 FORMAT (2I2,I5,F6.0,I5,I8)
    7 FORMAT (1H0)
    8 FORMAT (4Z4)
    9 FORMAT (5X,I4,4(5X,F8.5))
   11 FORMAT (1H1,'DATA ',I2,/)
   12 FORMAT (1H ,I5,' DATA POINTS'/)
   13 FORMAT (1H ,'SCAN RATE ',I5,' HERTZ'/)
   14 FORMAT (1H ,'COORDINATION NUMBER ',I8,/)
      J1   =   1
      JMAX =   1
      IDISK =   1
      IRATE = 1
      ICOORD = 000
      F1  =  1.
      IFNAME = 1
      PI  =  3.141592654
      WRITE (6,1)
```

```
      READ (5,2) IDISK
      WRITE (6,3)
      WRITE (6,4)
      WRITE (6,5)
      READ (5,6) IFNAME,JMAX,IRATE,F1,IR,ICOORD
C
C ** TRUNCATE DATA SET TO INTEGER NO. OF FUNDAMENTAL PERIODS **
C       IR  = NO. OF DATA RECORDS (OPTION SELECTABLE)
C       J1  = INITIAL DATA CHANNEL IDENT.
C      JMAX = FINAL DATA CHANNEL IDENT. (JMAX .GE.1 AND .LE.16)
C       F1  = FUNDAMENTAL FREQUENCY (HZ)
C      DELT = SAMPLE TIME FOR A DATA CHANNEL (SEC)
C     ICOORD= COORDINATION NO.
C       IP  = INTEGER NO. OF FUNDAMENTAL PERIODS
C        M  = INTEGER NO. SAMPLES FOR EACH CHANNEL (TRUNCATED FORM)
      N  =  IR/JMAX
       AN = N
      RATE = IRATE
      DELT = 1./RATE
       IP = IFIX(AN*F1*DELT)
       AP = IP
       M  = IFIX(AP/(F1*DELT))
C
C     NEXT READ IN SAMPLED DATA FROM DISK FILE
C
      DO 30  I = 1,M
         READ (IDISK,8) (IX(J,I), J = J1,JMAX)
   30 CONTINUE
C
C     SCALE INTEGER DATA AND CONVERT TO REAL NUMBERS
C
      DO  40   I = 1,M
         DO 35  J = J1,JMAX
              IF (IX(J,I).GT.2047)  GO TO 32
              AAA = IX(J,I)
              GO TO 33
```

```
32         AAA = IX(J,I) - 65536
33         CONST = 5./2047.
             X(J,I) = CONST * AAA
35      CONTINUE
40   CONTINUE
C
C
C     ECHO SCALED DATA VALUES TO CONSOLE
C
        WRITE (6,7)
        WRITE (6,11) IFNAME
        WRITE (6,12) IR
        WRITE (6,13) IRATE
        WRITE (6,14) ICOORD
        DO 45  I = 1,20
           WRITE (6,9)  (I,(X(J,I), J=J1,JMAX))
45   CONTINUE
C
C
C ** FIND CHANNEL BIAS AND R.M.S. **
C        Y(J) = AVE. VALUE OF CHANNEL "J"
C      RMS(J) = RMS VALUE OF CHANNEL "J"
C ** REMOVE BIAS FROM DATA **
50   DO 59 J=J1,JMAX
        AVE  = 0.0
51   DO 52 I=1,M
        AVE  = AVE + X(J,I)
52   CONTINUE
         AM  = M
        Y(J) = (1./AM)*AVE
         X2  = 0.0
53   DO 54 I=1,M
         X(J,I) = X(J,I) - Y(J)
         X2  = X2 + X(J,I)**2
54   CONTINUE
         X2  = (1./AM)*X2
```

```
      RMS(J) = SQRT(X2)
   59 CONTINUE
   65 WRITE(6,1000) J1,JMAX,ICOORD
      WRITE(6,1001) IR,DELT,F1
      WRITE(6,1002) M,N
      WRITE(6,1003)
   70 DO 71 I=J1,JMAX
      WRITE(6,1010) I,Y(I),RMS(I)
   71 CONTINUE
C ** FOURIER COEFFICIENT EVALUATION BRANCH **
C       KMAX = MAX. HARMONIC DESIRED
C      DELTAU= INTERCHANNEL SAMPLE DELAY (SEC)
C      X(J,I)= DATA ARRAYS (D.C. BIAS REMOVED)
C         J  = DATA CHANNEL, J1 TO JMAX
C         I  = DISCRETIZED SAMPLE INDEX, I=1 TO M
  100 DELTAU = 0.
       KMAX  = 2
  110 DO 123 K=1,KMAX
       AK  = K
       ARG = 2.*PI*F1*AK*DELT
       S1  = SIN(ARG)
       C1  = COS(ARG)
  115 DO 122 I=J1,JMAX
       AI  = (I-1)
       ARG = 2.*PI*F1*AK*(DELT + (AI*DELTAU))
       S2  = SIN(ARG)
       C2  = COS(ARG)
      A(K,I)=0.0
      B(K,I)=0.0
  120 DO 121 L=1,M
      A(K,I)= A(K,I) + X(I,L)*C2
      B(K,I)= B(K,I) + X(I,L)*S2
      AC2  = C2*C1 - S2*S1
      AS2  = S2*C1 + C2*S1
       C2  = AC2
       S2  = AS2
```

```
  121 CONTINUE
       AM  = M
      A(K,I)= (2./AM)*A(K,I)
      B(K,I)= (2./AM)*B(K,I)
      C(K,I)= SQRT(A(K,I)**2 + B(K,I)**2)
       A1  = ABS(A(K,I))
       B1  = ABS(B(K,I))
      IF(A1.LT.Ø.ØØ1.AND.B1.LT.Ø.ØØ1) GO TO 2ØØ
      PHI(K,I)= ATAN2(-B(K,I),A(K,I))*(18Ø./PI)
      GO TO 125
  2ØØ PHI(K,I) = Ø.Ø
  125 CONTINUE
  122 CONTINUE
  123 CONTINUE
  13Ø DO 137 K=1,KMAX
      WRITE(6,1Ø2Ø) K
  135 DO 136 I=J1,JMAX
      WRITE(6,1Ø25) I,A(K,I),B(K,I),PHI(K,I),C(K,I)
  136 CONTINUE
  137 CONTINUE
 1ØØØ FORMAT (1H1,4X,16HINITIAL CHANNEL:,T25,I2/7X,14HFINAL CHANNEL:,
     1 T25,I2/ 7X,14HCOORD. NUMBER:,T25,I8,//)
 1ØØ1 FORMAT(3X,18HTOTAL NO. SAMPLES:,T25,I5/
     1 2X,19HSCAN PERIOD (SEC.):,T25,E11.4/1X,2ØHREFERENCE FREQ (HZ):,
     2 T25,E11.4//)
 1ØØ2 FORMAT(1X,2ØHDATA PTS./CH., USED:,T25,I4,T35,7HAVAIL.:,T45,I4//)
 1ØØ3 FORMAT(5X,'SIGNAL BIAS AND R.M.S. VALUES',/
     1 2X,'CHANNEL',T15,'BIAS',T23,'R.M.S.'/)
 1Ø1Ø FORMAT(4X,I2,T12,F7.4,T22,F7.4)
 1Ø2Ø FORMAT(1HØ,4X,'FOURIER COEFFICIENTS FOR HARMONIC',I3/
     1 2X,'CHANNEL',T14,'COS',T24,'SIN',T34,'PHASE',T44,'MAG')
 1Ø25 FORMAT(4X,I2,T12,F7.4,T22,F7.4,T32,F7.2,T42,F7.4 )
  5ØØ FORMAT (1HØ,2X,'INDEX',T13,'X(1,I)',T23,'X(2,I)' /)
  5Ø1 FORMAT (4X,I3,T12,F7.4,T22,F7.4)
      STOP
      END
```

DATA 3

1024 DATA POINTS

SCAN RATE 300 HERTZ

COORDINATION NUMBER 911001

1	1.10650	1.09673	1.09428	1.08207
2	-0.05862	-0.06839	-0.07328	-0.08305
3	-1.21641	-1.22374	-1.23351	-1.23840
4	-2.14704	-2.15437	-2.15681	-2.16170
5	-2.69419	-2.69419	-2.69663	-2.69663
6	-2.76991	-2.76746	-2.76746	-2.76746
7	-2.36688	-2.36444	-2.35955	-2.35466
8	-1.53639	-1.52907	-1.52418	-1.51685
9	-0.42013	-0.41280	-0.40547	-0.39814
10	0.75721	0.76453	0.77186	0.77919
11	1.82218	1.82706	1.83195	1.83928
12	2.56961	2.57206	2.57694	2.57938
13	2.86517	2.86517	2.86517	2.86517
14	2.69907	2.69663	2.69419	2.69174
15	2.06400	2.05911	2.05178	2.04690
16	1.07474	1.06986	1.06253	1.05520
17	-0.09282	-0.09770	-0.10747	-0.11480
18	-1.24572	-1.25061	-1.26038	-1.26771
19	-2.16903	-2.17391	-2.17880	-2.18124
20	-2.70151	-2.70396	-2.70640	-2.70640

INITIAL CHANNEL: 1
FINAL CHANNEL: 4
COORD. NUMBER: 911001

TOTAL NO. SAMPLES: 1024
SCAN PERIOD (SEC.): 0.3333E-02
REFERENCE FREQ (HZ): 0.2000E 02

DATA PTS./CH., USED: 255 AVAIL.: 256

SIGNAL BIAS AND R.M.S. VALUES

CHANNEL	BIAS	R.M.S.
1	0.0445	2.0098
2	0.0444	2.0098
3	0.0444	2.0098
4	0.0442	2.0099

FOURIER COEFFICIENTS FOR HARMONIC 1

CHANNEL	COS	SIN	PHASE	MAG
1	1.8424	-2.1585	49.52	2.8379
2	1.8367	-2.1634	49.67	2.8379
3	1.8308	-2.1685	49.83	2.8380
4	1.8249	-2.1735	49.98	2.8381

FOURIER COEFFICIENTS FOR HARMONIC 2

CHANNEL	COS	SIN	PHASE	MAG
1	-0.0034	0.0091	-110.30	0.0097
2	-0.0031	0.0091	-108.86	0.0097
3	-0.0030	0.0092	-108.40	0.0096
4	-0.0031	0.0088	-109.56	0.0093

DATA 4

1024 DATA POINTS

SCAN RATE 3000 HERTZ

COORDINATION NUMBER 911002

1	-1.93698	-1.99804	-2.05178	-2.10308
2	-2.59648	-2.62335	-2.65022	-2.67465
3	-2.80166	-2.79922	-2.79433	-2.78700
4	-2.52565	-2.49389	-2.45725	-2.42062
5	-1.80019	-1.73913	-1.68051	-1.61700
6	-0.74499	-0.67171	-0.59599	-0.51783
7	0.42990	0.50562	0.58378	0.65950
8	1.54861	1.61456	1.68051	1.73913
9	2.40107	2.44260	2.48168	2.51832
10	2.83097	2.84074	2.85051	2.85540
11	2.78945	2.76991	2.75037	2.72594
12	2.28139	2.23253	2.18124	2.12750
13	1.37763	1.30923	1.23840	1.16756
14	0.23449	0.15633	0.08061	0.00244
15	-0.93307	-1.00879	-1.07963	-1.15535
16	-1.94187	-1.99804	-2.05178	-2.10308
17	-2.59892	-2.62579	-2.65266	-2.67709
18	-2.80166	-2.79922	-2.79433	-2.78700
19	-2.52076	-2.49145	-2.45481	-2.41573
20	-1.79531	-1.73669	-1.67318	-1.60967

```
   INITIAL CHANNEL:    1
     FINAL CHANNEL:    4
     COORD. NUMBER:     911002

  TOTAL NO. SAMPLES:    1024
 SCAN PERIOD (SEC.):    0.3333E-03
REFERENCE FREQ (HZ):    0.2000E 03

DATA PTS./CH., USED:    255      AVAIL.:    256

   SIGNAL BIAS AND R.M.S. VALUES
 CHANNEL     BIAS    R.M.S.

   1       0.0369    2.0113
   2       0.0362    2.0116
   3       0.0357    2.0114
   4       0.0353    2.0112
   FOURIER COEFFICIENTS FOR HARMONIC  1
 CHANNEL    COS        SIN        PHASE       MAG
   1      -1.0177   -2.6559     110.97     2.8442
   2      -1.0901   -2.6274     112.53     2.8445
   3      -1.1613   -2.5964     114.10     2.8443
   4      -1.2314   -2.5636     115.66     2.8440
   FOURIER COEFFICIENTS FOR HARMONIC  2
 CHANNEL    COS        SIN        PHASE       MAG
   1       0.0030    0.0011     -19.56     0.0032
   2       0.0036,   0.0009     -14.15     0.0037
   3       0.0040    0.0000      -0.64     0.0040
   4       0.0041   -0.0009      12.82     0.0042
```

DATA 5

1024 DATA POINTS

SCAN RATE 10000 HERTZ

COORDINATION NUMBER 911003

1	-1.21641	-1.59013	-1.89301	-2.16658
2	-2.48656	-2.63801	-2.74792	-2.80410
3	-2.78700	-2.70640	-2.57694	-2.40596
4	-2.02247	-1.73669	-1.41426	-1.06253
5	-0.45921	-0.07084	0.31265	0.69370
6	1.28236	1.62677	1.92721	2.19834
7	2.54274	2.69907	2.80410	2.85295
8	2.84074	2.76258	2.63556	2.46214
9	2.08109	1.79775	1.47289	1.12115
10	0.52760	0.14411	-0.23449	-0.62531
11	-1.21397	-1.55349	-1.85882	-2.13727
12	-2.48412	-2.63556	-2.74792	-2.80166
13	-2.78700	-2.70884	-2.57938	-2.40840
14	-2.02491	-1.73913	-1.41671	-1.06497
15	-0.46165	-0.07816	0.30288	0.68637
16	1.27992	1.61700	1.92233	2.19101
17	2.54274	2.69663	2.80410	2.85540
18	2.84074	2.76258	2.63556	2.46458
19	2.08598	1.79775	1.47777	1.12848
20	0.53493	0.13679	-0.24426	-0.62531

INITIAL CHANNEL: 1
FINAL CHANNEL: 4
COORD. NUMBER: 911ØØ3

TOTAL NO. SAMPLES: 1Ø24
SCAN PERIOD (SEC.): Ø.1ØØØE-Ø3
REFERENCE FREQ (HZ): Ø.1ØØØE Ø4

DATA PTS./CH., USED: 25Ø AVAIL.: 256

SIGNAL BIAS AND R.M.S. VALUES

CHANNEL	BIAS	R.M.S.
1	Ø.Ø313	2.Ø13Ø
2	Ø.Ø318	2.Ø129
3	Ø.Ø319	2.Ø13Ø
4	Ø.Ø316	2.Ø127

FOURIER COEFFICIENTS FOR HARMONIC 1

CHANNEL	COS	SIN	PHASE	MAG
1	Ø.54Ø9	-2.7947	79.Ø5	2.8465
2	Ø.1554	-2.8422	86.87	2.8464
3	-Ø.2294	-2.8373	94.62	2.8465
4	-Ø.614Ø	-2.7792	1Ø2.46	2.8462

FOURIER COEFFICIENTS FOR HARMONIC 2

CHANNEL	COS	SIN	PHASE	MAG
1	Ø.ØØ4Ø	Ø.ØØ17	-22.64	Ø.ØØ44
2	Ø.ØØ29	Ø.ØØØ5	-9.73	Ø.ØØ3Ø
3	Ø.ØØ31	-Ø.ØØ13	22.81	Ø.ØØ33
4	Ø.ØØ4Ø	-Ø.ØØ27	33.41	Ø.ØØ48

LIST OF REFERENCES

1. Englehardt, C. D., Data Acquisition System for Unsteady Aerodynamic Investigation, M. S. Thesis, Naval Postgraduate School, 1977.

2. Datel Systems, Inc., Sinetrac Series Model ST-800-32S/16D A/D Peripheral Systems Instruction Manual, Document No. ST8AMH3701, 1978.

3. Intel Corporation, MDS-800 Intellec Microcomputer Development System Hardware Reference Manual, 1975.

4. Teletype Corporation, Model 40 Installation and Servicing Manual, Manual 347, Issue 3, 1976.

5. International Business Machines, Component Description: IBM 2701 Data Adapter Unit, File No. 2701-09, 1971.

6. Intel Corporation, SBC 534 Four Channel Communications Expansion Board Hardware Reference Manual, Manual No. 9800450A, 1977.

7. Intel Corporation, 8080 Microcomputer Systems User's Manual, 1975.

8. Intel Corporation, Peripheral Design Handbook, 1978.

9. Intel Corporation, 8080 Assembly Language Programming Manual, 1976.

10. Digital Research Corporation, An Introduction to CP/M Features and Facilities, 1976.

11. Casko, J. D., A Microprocessor Controlled Automatic Data Logging System (ADL), M. S. Thesis, Naval Postgraduate School, 1977.

12. Adam Osborne and Associates, Inc., An Introduction to Microcomputers, 1975.

13. Pickelsimer, B. M., Data Reduction for the Unsteady Aerodynamics on a Circulation Control Airfoil, M. S. Thesis, Naval Postgraduate School, 1977.

INITIAL DISTRIBUTION LIST

		No. Copies
1.	Defense Documentation Center Cameron Station Alexandria, Virginia 22314	2
2.	Library, Code Ø142 Naval Postgraduate School Monterey, California 9394Ø	2
3.	Department Chairman, Code 67 Department of Aeronautics Naval Postgraduate School Monterey, California 9394Ø	1
4.	Professor L. V. Schmidt, Code 67sx Department of Aeronautics Naval Postgraduate School Monterey, California 9394Ø	5
5.	Kathryn Strutynski, Code Ø141 W. R. Church Computer Center Naval Postgraduate School Monterey, California 9394Ø	1
6.	LT Mack Taylor Elliott, USN Air Anti-Submarine Squadron Forty-One Naval Air Station, North Island San Diego, California 92135	1

Thesis 178459
E3687 Elliott
c.1 High speed data
acquisition system.